The Patent Office

INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

JC903 U.S. PTO
09/661463
09/13/00

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed M Coakin

Dated 1 September 2000

Patents Form 1/77

Patents Act 1977
(Rule 16)

THE PATENT OFFICE
A
19 MAY 2000
NEWPORT

The Patent Office

1/77

19MAY00 E538031-1 D01463
P01/7700 0.00-0011986.7

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office
Cardiff Road
Newport
Gwent NP9 1RH

1. Your reference — 30990066 GB

2. Patent application number *(The Patent Office will fill in this part)* — **0011986.7** — 19 MAY 2000

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

   Hewlett-Packard Company
   3000 Hanover Street
   Palo Alto
   CA 94304, USA

   Patents ADP number *(if you know it)* 496 588004

   If the applicant is a corporate body, give the country/state of its incorporation — Delaware

4. Title of the invention — Auction Method and Apparatus For Electronic Commerce

5. Name of your agent *(if you have one)*

   "Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

   Richard A. Lawrence
   Hewlett-Packard Ltd, IP Section
   Filton Road
   Stoke Gifford
   Bristol BS34 8QZ

   Patents ADP number *(if you know it)* 7563083001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*
   a) any applicant named in part 3 is not an inventor, or
   b) there is an inventor who is not named as an applicant, or
   c) any named applicant is a corporate body.
   See note (d))

   Yes

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	40
Claim(s)	5
Abstract	1
Drawing(s)	20 +20

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	
Request for preliminary examination and search *(Patents Form 9/77)*	
Request for substantive examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	

11. I/We request the grant of a patent on the basis of this application.

Signature Richard A Lawrence Date 18/5/00

12. Name and daytime telephone number of person to contact in the United Kingdom

Janet Smith, 0117-312-8026

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) *If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.*

b) *Write your answers in capital letters using black ink or you may type them.*

c) *If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.*

d) *If you have answered 'Yes' Patents Form 7/77 will need to be filed.*

e) *Once you have filled in the form you must remember to sign and date it.*

f) *For details of the fee and ways to pay please contact the Patent Office.*

AUCTION METHOD AND APPARATUS FOR ELECTRONIC COMMERCE

Field of the Invention

The present invention relates to computing entities configured for electronic commerce, and particularly although not exclusively, to a method and apparatus for making bids and offers at electronic auctions over the internet.

Background to the Invention

It is known on the internet that several commercial websites are present which serve the purpose of auctioning goods. Referring to Fig. 1 herein there is shown an example of a prior art system in which first and second commercial auction websites 100, 101 are each supported by a corresponding computer entity 102, 103 operating on a website server. A user of the auction websites over the internet 104 accesses the auction websites 100, 101 by use of his own computing entity 105. In Fig. 1, auction websites 100, 101 and the user are shown as logical entities (being websites supported by the computing entities) and the user operates a browser software 106 resident on the user's computer entity 105. Examples of such auction websites include those provided by eBay™ and QXL. A human user having a conventional personal computer equipped with a conventional browser software may use the browser software to access the auction websites over the internet in known manner, and using the browser may generate a screen display downloaded from the website which provides the user with an interactive interface on his computer entity 105 through which the user can participate in an auction hosted by the website servers 102, 103. To participate in an auction, a user typically must register with the auction site server by entering the user's details such as company name, personal name, credit card number, and address, in to dialog boxes provided on the website interface. The user accesses information concerning goods sold through the auction website.

Referring to Fig. 2 herein, there is illustrated schematically aspects of a typical user interface display of a known auction website, which a human user can access on his or her graphical user interface by pointing a browser software at the website. Typically such a display comprises a listing of goods or services to be sold, a quantity of goods or services, a current highest bid price which is being bid for the goods or services, a number of bids which have been received by the website for those goods or services, a time at which the auction started, and a time at which the auctioning of the goods or services may close, as well as other details concerning payment method and shipping terms.

Referring to Fig. 3 herein there is illustrated an example of a prior art screen display accessed by a user for making a bid. A user enters an amount which the user bids for the goods using a dialog box provided. The bid is relayed over the internet to the website, which inputs the bid information into its controlling software to determine whether or not the bid is successful. Once the auction software accepts a successful bid, a confirmation message is sent to the computing entity of the successful bidder, and a transaction, including supply of goods and payment for goods, occurs by conventional means, for example postal delivery of goods and conventional credit card transaction clearance.

It is known in prior art systems to provide at a known auction website a software functionality known as a 'bidding elf', which is provided as part of an auction website, and which is capable of bidding in an auction on behalf of a buyer. The bidding elf provides a user with a screen display having dialog boxes into which the user can enter an upper maximum limit to which he or she is prepared to bid for various goods. The bidding elf then places bids on behalf of the user, varying the bid level incrementally up to the maximum predetermined limit set by the user. The bidding elf removes the need for the user to sit at his visual display continuously to monitor current bidding whilst the auction is occurring and continue to increase his bid incrementally up to his maximum bid.

With known electronic auctions, generally a user can only attend one auction at any one time, since the website browser can only be pointed at one auction at any one time and the user interface of one auction website can be displayed on a single monitor at any one time. If a user wishes to attend more than one auction simultaneously, in order to source goods or services at the best price from several auctions, then the user must arrange a plurality of website browsers supported by one or more computing entities to communicate with the plurality of auctions, and must visually monitor each separate auction site simultaneously. Such a system is possible, using conventional technology, but is inelegant, requiring several items of computing equipment by the user, and requiring manual monitoring and interaction of several auction sites at once. The prior art bidding elf can only make bids in one auction, being the auction website with which it is associated. Different prior art elves at different websites cannot communicate with each other to coordinate bidding in several auctions.

Summary of the Invention

One object of the present invention is to provide a method and apparatus to enable a user to participate in several electronic auctions simultaneously in a controlled and coordinated manner to obtain a best price contract for goods and services from a plurality of auctions. In one implementation, a user may set an upper and/or lower price limit for bid or offer of goods or services at a plurality of auctions, and thereby participate in risk free buying or selling of goods and services. The user may be guaranteed that the maximum price bid will not be exceeded, or a minimum price offered will not be exceeded in specific embodiments of the invention.

Specific embodiments may enable a user to obtain items at a preferential rate in one auction whilst participating in one or more other auctions for the same goods/services and whilst avoiding buying of more goods than required, or selling

more goods than required through participating in other ones of the plurality of auctions.

According to first aspect of the present invention there is provided an electronic trading entity comprising:

a processor;

a memory means;

a communications means;

monitoring means for monitoring a plurality of trading data displayed by a plurality of auction entities;

quantity calculation means for calculating quantities of items for trading with said plurality of monitored auction entities,

price calculation means for calculating optimal prices of items to be traded with said plurality of auction entities;

wherein said price calculation means operates to process said trading data to obtain trading prices for a quantity of items calculated by said quantity calculation means, said prices calculated for a set of chosen auction entities selected from said set of all said plurality of auction entities;

said price calculation means operating to process said selected data to obtain trading prices for a predetermined amount of items across all said selected set of said plurality of auction entities.

Preferably the electronic trading entity further comprises search means for searching electronically for individual ones of said auction entities.

Preferably said amount calculation means comprises:

means for storing bid data describing a plurality of bids in at least one of said plurality of auction entities;

means for storing user bid data describing a number of bids placed by said trading entity with at least one of said plurality of auction entities; and

means for determining an optimum number of bids and corresponding price amounts of said bids to place with at least one of said plurality of auction entities.

According to a second aspect of the present invention there is provided a method of operating an electronic trading entity for trading a plurality of tradable items, said method comprising the steps of:

monitoring a plurality of auction entities;

inputting a plurality of trading data from said plurality of auction entities;

processing said trading data of said plurality of auction entities to determine an optimum set of amount data describing quantities and prices of said tradable items for trading by said trading entity.

Said step of processing trading data of said plurality of auction entities may comprise:

storing in memory a plurality of bid data displayed by said plurality of auction entities;

determining a number of active bids of said trading entity already sent by said trading entity to at least one of said plurality of auction entities;

constructing a plurality of currently successful sets of said bid data of said plurality of auction entities;

determining a cost to outbid each said set of currently successful bids;

selecting a said currently successful set of bids having an optimum cost to outbid; and

constructing a set of bids of said trading entity which are higher than corresponding bids of said optimum cost currently successful bid set.

Said step of constructing a set of bids of said trading entity which are higher than said optimum cost currently successful bid set may comprise:

for each bid of said optimum bid set, said bid made either by said trading entity or a third party, selecting a highest third party bid appearing in an identical auction entity to said bid, which also appears in said optimum bid set; and

generating a trading entity bid of an amount equal to said highest third party bid plus a minimum bid increment.

Said step of processing said trading data of said plurality of auction entities may comprise:

storing in memory a plurality of offer data of said plurality of auction entities;

determining a number of active offers already placed by said plurality of auction entities;

constructing a plurality of sets of offers of said auction entities;

determining a value of each of said set of auction entity offers;

selecting a set of auction entity offers having an optimum value to undercut with new offers;

determining whether said value is within a predetermined value limit; and

generating a set of offer data matched to said optimum set of offer data.

Said step of processing data may comprise:

storing a plurality of offers of a plurality of auction entities,

selecting a plurality of sets of said plurality of offers, each set having a number of members equal to a number of items to be obtained by said trading entity;

for each said selected set, determining a value of said set;

selecting a said set having a maximum value;

generating a set of offers which correspond with said set having said maximum value.

Said step of generating a set of offers may comprise:

for each offer of said set of optimum offers, generating an offer having value equal to the corresponding respective offer value, minus a predetermined amount.

According to third aspect of the present invention there is provided a method of exchanging data between an electronic trading entity and a plurality of electronic auction entities, said method comprising the steps of:

said trading entity monitoring data displayed by said plurality of auction entities;

said first trading entity extracting from said monitored data, data describing prices of individual items to be traded;

said first trading entity determining from said price data an optimum set of trading data for sending to said plurality of auction entities; and

said first trading entity communicating said trading data to said plurality of auction entities.

Brief Description of the Drawings

For a better understanding of the invention and to show how the same may be carried into effect, there will now be described by way of example only, specific embodiments, methods and processes according to the present invention with reference to the accompanying drawings in which:

Fig. 1 illustrates schematically an example of a prior art system in which first and second commercial auction websites are each supported by corresponding computer entity;

Fig. 2 illustrates schematically an example of a typical prior art user interface display of a known auction website;

Fig. 3 illustrates schematically further features of the typical prior art user interface display of the known auction website;

Fig. 4 illustrates schematically operation of a trading entity according to first specific implementation of the present invention in an internet environment;

Fig. 5 illustrates schematically an architectural structure of the trading entity of Fig. 4;

Fig. 6 illustrates schematically interaction of the trading entity with a plurality of auction entities over the internet and interaction of the trading entity with a user;

Fig. 7 illustrates schematically a logical layout of the trading entity of Fig. 4;

Fig. 8 illustrates schematically process steps carried out by the trading entity for procuring a single good or item from a plurality of auction entities;

Fig. 9 illustrates schematically process steps carried out by a monitoring and trading component of the bidding algorithm for bidding for several goods or services at once from a plurality of auction entities;

Fig. 10 illustrates schematically a data table recording data describing a plurality of bids and/or offers which the trading entity is actively pursuing at a plurality of auction entities;

Fig. 11 illustrates schematically a bid selection algorithm for selecting a set of bids to place across a plurality of auction entities;

Fig. 12 illustrates schematically an array of bids for each of a plurality of N auctions in which individual sets of bids D are selected;

Fig. 13 illustrates schematically an example of a specific set of bids selected exclusively from a single auction;

Fig. 14 illustrates schematically a set of bids comprising 4 bids, placed in 2 auctions at levels which outbid currently active bids of third parties in those auction entities;

Fig. 15 illustrates schematically an offer selection algorithm for selecting a set of offers to be made in a plurality of auction entities;

Fig. 16 illustrates schematically a method for determining an optimum set of bids to place with one or a plurality of auction entities;

Fig. 17 illustrates schematically a set of bids which may be displayed by a plurality of auction entities and stored by a trading entity according to a specific embodiment of the present invention; and

Fig. 18 illustrates schematically a set of offers which may be generated by the trading entity, to match with a set of bids as illustrated in Fig. 17 herein.

D tailed Description of the Best Mod for Carrying Out the Invention

There will now be described by way of example the best mode contemplated by the inventors for carrying out the invention. In the following description numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent however, to one skilled in the art, that the present invention may be practiced without limitation to these specific details. In other instances, well known methods and structures have not been described in detail so as not to unnecessarily obscure the present invention.

A first specific implementation according to the present invention is concerned primarily with electronic commerce according to the 'English auction' system in which a seller takes competing bids from a plurality of bidders, with a current highest bid price from one of the bidders being the price which other bidders must exceed in order for a successful transaction to occur. Where the highest bid price is not exceeded for a predetermined period, the seller may choose to either withdraw the goods from auction if his reserve price is not met, or the seller may enter into a transaction with the bidder making the highest bid in the auction.

Throughout this description, transactions concerning goods or services will be referred to. Where a transaction to buy or sell a good is mentioned, it will be understood that the subject of the transaction may be either a good or a service or both a good and service, the terms good and service being used interchangeably, and similarly a reference to a transaction involving a service will be construed accordingly to include a transaction involving a good.

In this specification the term "lot" is used to refer to a minimum quantity which is subject to a single transaction. A number of individual goods or services comprising a lot may be defined by a user of a computer trading entity according

to specific implementations of the present invention described herein or may be specified by an entity with whom the computer trading entity trades. A user may define a lot size for any particular type of good or service as being one, or a preset plurality of goods or services.

Referring to Fig. 4 herein, there is illustrated schematically an interaction of a trading device 400 according to a first specific implementation of the present invention with a plurality of auction entities 402 – 404. The trading device comprises a computer program operating to control a first computing entity 401 comprising a data processor and memory means. Each auction entity comprises a computer program operating a corresponding computing entity 405 – 407, the computing entities each comprising a data processor and associated memory means, the trading device 400 communicating with the one or plurality of auction entities over the internet 408.

Referring to Fig. 5 herein, there is illustrated schematically an architectural structure of trading device 400. The trading device comprises a data processing means 500; an area of memory 501 for storage of data and instructions used by processor 500; a user interface 502, comprising a video display screen, keyboard and pointing device, for example a mouse device; an operating system 503, for example the known Unix™ operating system, Microsoft Windows™. operating system, or Linux™ operating system; a modem 504 for communicating with other internet devices; a plurality of applications 505, including a web browser 506, and a set of algorithms 507 for placing bids and offers at auction sites, the algorithms 507 co-operating with web browser 506. The web browser 506 may comprise a conventional web browser, for example the known Microsoft Internet Explorer™ browser.

Referring to Fig. 6 herein, there is shown schematically communication of trading device 400 with the plurality of auction entities, and with a user of the

device. Trading device 400 receives hypertext mark-up language (HTML) or XML signals 600 from the auction entities, and sends data 601, also in HTML or XML to the auction entities over the internet. Trading device 400 communicates with the user interface using HTML signals, and can receive data input and commands from a user via the user interface, for example by the user typing in commands, or using the pointing device.

The trading device comprises a local set of software commands on the computer entity 401, which operate to parse information from the auction entities, and receives HTML signals, in the same way as the web browser. The trading device extracts information from the HTML signals concerning a current status of an auction entity, including information on a highest current bid and the bidding increments set by the auction entity.

The trading device 400 contains a plurality of algorithms for interacting with the user and one or more auction entities as will now be described.

Referring to Fig. 7 herein, there is illustrated a logical layout of trading device 400 including arrows indicating flows of data between logical components of the trading device 400. Trading device 400 comprises a web browser 506; monitoring and trading algorithms 700; one or more price calculation algorithms 701, an auction selection algorithm 702, and a URL store 703. Web browser 506 may monitor one or a plurality of auction entities under control of monitoring and trading algorithm 700. Monitoring and trading algorithm 700 interacts with price calculation algorithm 701 to obtain information concerning calculated optimum prices and lowest/highest prices. Price calculation algorithm 701 communicates with auction selection algorithm 702 for selecting auction entities having the best prices for selected goods or services. URL store 703 stores URL addresses of selected auction entities which are referred to by the auction selection algorithm 702 and monitoring and trading algorithm 700. User interface 502 communicates

with web browser 506, and monitoring and trading algorithm 700, so that a human user of the trading device can monitor and participate in trades carried out by the monitoring and trading algorithm 700 and actively search and select auction entities using web browser 506. Each of monitoring and trading algorithm 700 and web browser 506 are capable of communicating with auction entities via the modem facility 504 of the host computer entity, via the internet.

Whilst communications between the trading device and the auction entities are described herein as being carried out in HTML or XML, protocols operated by the trading device 400 for communication in the general sense is not limited to these languages, and in the general case communication may occur by other standard interfaces, for example FIPA (foundation of intelligent physical agents) defined languages.

In some embodiments, algorithms for communicating with agent entities may reside within a web browser, in which case the algorithms may generate HTML driven screens which the user may review using his conventional web browser. Information which the trading device displays to the user includes information concerning the status of a set of auction sites selected by the user. The trading device may be directed to web sites by a user typing in addresses of known websites, using the URL of the websites, or alternatively a communication algorithm within the device may be configured to use the browser to search for auction sites by entering in a keyword search via the browser, into a conventional search engine, e.g. Yahoo™ or Alta Vista™.

A user of the trading device needs to have confidence in the integrity and reliability of the auction entities which the trading device is communicating with. Consequently, in the best mode implementation, rather than leave selection of auction entities to the trading device, a user may select a plurality of auction entities with which the trading device communicates, for example by inputting the

URL data of selected auction entities to the trading device, which the trading device stores in URL memory store 703. Other user inputs to the trading device include:

- A list of goods or services which the user wishes to buy, sell or monitor for. Goods need not necessarily be actively bought or sold, but can be monitored and a user alerted to any particular goods of interest when these arise on an auction site.

- Information concerning auctions which the user wishes to participate in. This information may include a general instruction to participate in any auction which the user has pre-selected and has entered its URL into the trading device.

- A quantity of goods or services to buy or sell. This data may comprise a maximum quantity of goods or services to buy or sell, and a minimum quantity of goods or services to buy or sell. For example, when buying or selling electronic components, lots of 1000 units are common, and lots of 100 units or lower are generally not traded.

- In the case of a buy instruction from the user to the trading device, information describing an upper bid price limit (and currency type) which the user is prepared to pay without the trading device referring back to the user.

- In the case of buying goods or services, an absolute upper bid price limit and currency above which the user is unwilling to purchase the goods or services.

- In the case of selling goods or services, a lower offer limit below which the user requires referral of price information before selling the goods or services.

- In the case of selling goods or services, an absolute lower offer limit below which the user is unwilling to sell the goods or services.

If the user is buying, and the goods are within the user's upper bid limit, the trading device may buy goods and services automatically without further referral to the user. Similarly, if the user is selling goods and services and the prices obtained are above the user's lowest offer limit, the trading device may sell goods and services automatically without further referral to the user.

Referring to Fig. 8 herein, there is illustrated steps of a simple algorithm for purchasing one lot of goods or service only from one of a plurality of N auction entities.

In step 800, the user inputs a maximum price limit P_{MAX} being a price which should never be exceeded in any of the N auctions for a quantity of goods or services.

In step 801, the user inputs a maximum price data P_{MAX} into his user interface, which is received by trading device 400. In step 801, the user enters a referral price $P_{REF,}$ which is a price above which any bids to be made by the device must be referred back to the user before proceeding. In step 802, the trading device monitors a plurality N of pre-selected auction entities, which have previously been selected by the user, and which are selling the goods or services of interest to the user. In step 803, the trading device monitors the current highest bids b in each of the plurality of N auctions monitored. This results in a plurality of bids b_{1i}, ... b_N, being the current highest bid operating in each of the N auctions. In step 804, if the trading device has a currently entered bid in any of the N auctions, which is the highest bid in that auction, that is to say if the trading device is actually leading the bidding in any one of the plurality of N auctions, then the algorithm idles, and continues to monitor the N auctions and current bid

prices in steps 802, 803 as described previously. However, if the trading device is not currently leading the bidding in any one of the N auctions, in step 805 the algorithm selects the auction having the lowest current active bid (the active bid being the highest bid in that auction) in step 805. In step 806, it is checked whether there is more than one auction having an equivalent lowest active bid. If there is only one such auction, then in step 809, the algorithm calculates a bid at the level $b_i + \delta$ in the selected auction i. However, if there is more than one auction having a lowest active bid price, that is to say that there is a same lowest active bid price for the goods or services in more than one auction, then the algorithm selects from these the auction having the earliest finish time, and the lowest active bid price in step 807. In step 808, the algorithm then proceeds to calculate a bid $b_i + \delta$ in the selected auction i, where δ is the smallest bid increment acceptable in that auction i. In step 810 the algorithm compares the bid $b_i + \delta$ with the price limit above which all bids must be referred to the user P_{REF}. If the bid $b_i + \delta$ is below the referral limit P_{REF}, in step 811, the algorithm proceeds to place the bid $b_i + \delta$ in the selected auction and returns to monitoring state 802. However, if the bid $b_i + \delta$ is above the limit P_{REF} above which all bids must be referred to the user, in step 812, the algorithm compares the bid $b_i + \delta$ with the absolute maximum price limit P_{MAX} set by the user to see if it is below P_{MAX}. If the calculated bid exceeds P_{MAX} then the algorithm stops and no further bidding takes place. The algorithm stops in step 813. However, if the calculated bid $b_i + \delta$ is within the maximum upper price limit P_{MAX}, in step 814, the algorithm refers the bid to the user for authorization. If the user authorizes the bid in step 815, then the bid $b_i + \delta$ is placed in the selected auction in step 816 and returns to monitoring step 802. However, if the user does not authorize the bid, or rejects the bid $b_i + \delta$ then in step 817, the algorithm stops and no further bidding takes place.

The above algorithm ensures that the user always has a leading bid in one of the plurality of N auctions, placing the lowest possible bid to take a lead in one

of the plurality N auctions and without placing a bid in any other auction than the one in which the lowest possible bid is placed until that bid is overtaken. When the user's bid is overtaken in the auction in which the user is bidding, the algorithm selects the next auction having the lowest active bid price and places a bid in that auction. This auction may be the same auction as the one in which the user has just been outbid in, or may be a different auction of the plurality N auctions. The algorithm is always searching for the lowest current active bid which allows the user to lead in one of the plurality N auctions for the required single lot of goods or services, and always below the user's referral price limit P_{REF} which the user is prepared to pay without the trading device referring back to the user.

The referral mechanism provided where a bid is above the user's price limit above which all bids need to be referred back to the user P_{REF}, but below the user's absolute maximum price limit P_{MAX} above which the user will never want to bid for the lot of goods or services, is provided primarily to give the user confidence in the trading device and gain acceptance of the bidding system from the user. However, an optimally behaving user should not use the referral mechanism, but should set his maximum price limit above which he would never wish to pay to be the same as the price limit above which referrals should be made, that is to say $P_{REF} = P_{MAX}$. The trading device will never pay more than the upper limit set by the user, and will not reveal that upper limit to the auction, or to competing bidders.

Referring to Fig. 9 herein, there is illustrated schematically an example of procedure steps carried out by monitoring and trading algorithm 100 for bidding for several lots of goods or services at once at a plurality of auction entities. A number M goods of a specified type are to be purchased from a plurality of N auctions at an optimal overall price, that is to say, the best prices available throughout the N auctions. Using this algorithm, some of the plurality M goods

may be bought at different prices to others, but overall, the best prices for the M goods are obtained, subject to availability, at the plurality N of auctions. In each of the auctions, different bidders may bid a different price per good for different lots of goods where the price depends on the quantity traded. For example, a first bidder may bid say $500 each for 5 items (5 lots) whereas in another auction, another bidder may bid $450 per item for 10 items (10 lots) of the same goods. The algorithm of Fig. 9 deals with the feature of the bid price per lot being dependant upon the size of transaction by treating each of the M lots to be traded separately, and attaching a separate bid price to each one of the plurality M lots to be bid for. Therefore, in the algorithm, instead of bidding $2,500 for 5 lots, the algorithm breaks the transaction down into 5 individual bids of $500 for each one of 5 lots, resulting in 5 bids instead of 1 bid in this example.

In step 900, the user selects each of the plurality of N auctions in which the user wishes to bid for the goods or services. This may be achieved by selecting the auctions using a conventional web browser and inputting the URL of each selected auction into the trading device. In step 901, the user inputs the single type of good or service, and the total number of lots M of the good or service to be bid for in the plurality N auctions via the user interface 502. In step 902, the user inputs an absolute maximum bid price P_{MAX} per lot, which the user is prepared to pay. A lot of goods comprises a predetermined quantity of goods or services which the user is buying, and may be a single good, or a plurality of goods, for example 1000 individual items. In step 903, the user inputs a referral price, P_{REF} being a cost amount per lot above which any bids must be referred to the user before they can be entered at any of the auctions. As described above, a user who behaves optimally will set $P_{REF} = P_{MAX}$ so that in practice no referral of price occurs back to the user. In step 904, each one of the plurality of N auctions are continuously monitored to check the current active bid prices, where the active bid price is the highest bid price currently operating by any bidder in that individual auction. In general, the current active bid price in each of the plurality

of auctions will differ, but in a theoretically 100% efficient marketplace, the currently active bid price in all auctions may be the same. In step 905, for each auction i the algorithm considers the top M_i bids

$b_1^i, \ldots, b_{Mi}^i$

where $b_1^i \geq b_{Mi}^i$

For each auction, the top M_i bids are stored in a table, with the highest bid price (the highest currently active bid price which would be successful in any auction) b_1^i at the top of the data table and the lowest active bid price b_{Mi}^i at the bottom of the table, b_{Mi}^i being the lowest bid price in the auction which would succeed in buying the goods or services. Further bids in the auction, which are currently unsuccessful (inactive) are labeled b_{mi+1}^i, b_{mi+2}^i … and so on.

The monitoring and trading algorithm maintains a database in memory in which for each of the N auctions which are being monitored, there is stored data describing a price amount for a plurality of individual bids which the algorithm has placed and which are currently active. An active bid is defined herein as a bid which has been placed with an auction entity, and which is of an amount sufficient to secure a trade for that lot of goods or services in the absence of any subsequent higher bids for those same goods or services, and disregarding whether the amount exceeds a reserve price limit set by the auction entity for that same lot of goods or services.

The algorithm may also store connected with the active bid information for each lot, further information describing the proposed trade, including for example a description of the goods or services for which the bid is being made, a quantity of individual items for which the bid is being made, that is to say the number of individual items in a lot which is being bid for, a close time by which the bid will be accepted or rejected by the auction entity, and a quality information describing a

quality of goods or services to be provided. Examples of the types of information which may be stored by the algorithm are illustrated in Fig. 10 herein. The data stored may comprise a bid identifier data 1000 being a unique identification code generated by the bid algorithm for a particular bid which has been placed; an auction identification data 1001, comprising a URL of the auction in which the bid identified by corresponding respective bid identifier 1000 has been placed; a lot identification data 1002, indicating a specific lot of goods or services for which the bid has been placed; a description information 1003 comprising a brief description of the goods or services for which the bid has been placed, or alternatively comprising a pointer to a separate table in which a more detailed description may be stored; a lot size, comprising the number of individual items (goods or services) in the lot which has been bid for; a monetary amount bid 1005, including a currency indicator relating to the currency in which the bid has been made, a time and date data 1006 at which the bid was made; a closing time of the auction for the particular lot which has been bid for; and a status indicator 1008 describing whether the current bid is in an active or inactive state.

In step 907, the monitoring and trading algorithm sums the number of bids, being one bid per specified lot, for the specified goods/services of the same type currently active, in all the selected auctions. For example, referring to Fig. 10, there are 5 bids active for lots of ten thousand 10k resistors, all of which are currently active, and spanning two individual auctions. In this example, the lot size is ten thousand individual resistors, and the price per lot which has been bid varies from \$25 to \$28. If L_i is the number of bids currently active which have been placed by the bidding device (in the example of Fig. 10 for the 10k resistor lots), L_i equals 5, then if for all auctions i = 1 to N, then if the total number of bids L_i is greater than or equal to M as determined in step 908, then no further bids are placed and the algorithm continues to monitor each of the plurality of N auctions in step 904 and continues with subsequent steps 905 – 907. However, in step

908 the number of current active bids for all auctions is less than the total number of lots of goods or services to be obtained, that is to say:

Equation 1

$$\sum_{i=1}^{N} L_i \leq M$$

Then, the algorithm places additional bids according to a bid selection algorithm hereinafter described with reference to Fig. 11, so that an additional Q active bids are held by the trading entity, where Q is determined in equation 2 below.

Referring to Fig. 11 herein, there is illustrated an auction selection algorithm for selecting a subset of the plurality N of selected auctions in which to place new bids. The algorithm in Fig. 11 operates in real time, continuously inputting new bid values from each of the plurality of N auctions as they occur, and placing new bids on an ongoing basis as bids which have previously been placed by the algorithm are overridden by competing bids from third parties.

An object of the algorithm is to determine the lowest cost bids to place in the plurality of auctions N, in order to obtain the required number of lots of goods or services 1100. In step 1101, the algorithm receives input of data describing the current active bids of all of the plurality of N selected auctions. These bids comprise all the bids which, in the absence of any new bids, will be successful in obtaining goods or services (subject to these bids not being below a reserve price set by the auction). These bids are shown schematically in Fig. 12 herein as an array of bids for each auction i of the plurality N auctions. For the purpose of clarity in this example diagram, we assume every auction is selling exactly 4 lots.

Equation 2

$$Q = M - \sum_{i=1}^{N} L_i \le M$$

The trading device then determines the optimum least cost bids to place which will result in an additional Q active bids.

In each auction A, there are a plurality of bids which need to be beaten in order to be successful in acquiring a given number of goods j from that auction. A plurality of potential sets of bids are constructed to outbid existing bids in the plurality N of auctions as illustrated in step 1102 herein. The beatable – j set of existing bids in auction A is defined as the set

Equation 3

$(b_{M_i}{}^{i}{}_{-j+1}, \ldots b_{M_i}{}^{i})$ for $j \ge 1$

This is the set of number j lowest bids which would succeed if the auction were terminated immediately. For example, in Fig. 12, 1201 shows the beatable – 1 set for auction 1, and 1202 shows the beatable – 2 set for auction 3.

The beatable zero (beatable – 0) set of bids for any auction is defined to be an empty set of bids.

A plurality of sets of bids can be formed by taking the members of exactly one beatable – j set ($j \ge 0$) from every auction A_i. The set of all sets which can be formed by taking the members of exactly one beatable – j set ($j \ge 0$) from every auction A_i is defined as the set Φ. In Fig. 12 there are illustrated 3 members of Φ, each with 3 elements;

- The set made from the beatable – 1 set of auction 1 (1202), and the beatable – 2 set of auction 3, and the beatable - 0 sets of all other auctions.

• The set (1203) made of the beatable – 3 set of auction N, and the beatable – 0 sets of all other auctions.

• The set (1204) made up of the beatable – 1 sets of auctions 1, 2 and 3 and the beatable – 0 set of all other auctions.

All sets D in the set Φwhich do not contain exactly Q bids made by parties other than the trading device are deleted.

(1103) - For each set $D \in \Phi$ a cost of the set D is defined

Equation 4

$$\text{Cost}(D) = \sum_{i=1}^{N} j_i (b^{i}_{m_i - j_i + 1} + \delta_i) - \sum_{b \in D_A} b$$

Where D_A is the set of bids in set D made by the trading device and j_i is the number of the highest bid which has been chosen by the algorithm in a particular set of bids of a selected auction A_i.

Equation 4 calculates the cost of outbidding each beatable – j set in D by placing j_i new bids of value $b^{i}_{m_i - j_i + 1} + \delta_i$, and subtracts the cost of existing bids in these sets made previously by the bidding device. This is illustrated schematically in Fig. 13, for a simple case where Q = 4, and the set D comprises the highest 4 bids in a single auction. In order to be successful in auction M, 4 bids 1302 must be placed, which exceed the highest bid b_1^M in auction M. The cost of placing these bids = $4(b_1^M + \delta)$, where δ is the minimum bid increment allowable in the auction. Similarly, in another example illustrated schematically in Fig. 14 herein, there is shown another set D_{pr} comprising a number Q = 4 bids, placed in order to outbid the currently active bids of third parties in auctions P and

R. To be successful in auctions P and R, 4 bids need to be placed 1501, 1502, being 2 bids of an amount $b^P_3 + \delta$ in first auction P, and 2 bids each at an amount $b^r_3 + \delta$ in auction R, giving a total cost $2(b^P_3 + b^r_3 + \delta_P + \delta_R)$ for the bid set D_{PR}, where b^P_3 is the second lowest currently active bid price in first auction P, δ_P is the minimum bid increment in first auction P, b^r_3 is the second lowest currently active bid price in second auction R, and δ_R is the minimum bid increment in second auction R.

δ_i is the bid increment, which is specified by a particular auction. Bid increment data is obtained by a parser in the monitoring algorithm which reads the bid data directly from the auction entity, the data being transferred in a language such as HTML or XML.

(1104) - A set of bids having a minimum cost D_{MIN} is obtained by choosing $D_{MIN} \in \mathbb{C}$ such that

Equation 5

$$\text{Cost}(D_{MIN}) \leq \text{Cost}(D)$$

$$\forall D \in \Phi$$

If more than one set satisfies equation 5, then the latest auction finish times in each set are considered. The set with the earliest such finish time is chosen.

In step 1105 the algorithm then proceeds to deploy those bids in the relevant auctions at the appropriate calculated levels, as long as those bids are below the referral price P_{REF} set by the user. Any new bids which are calculated by the algorithm which are above the referral price P_{REF} will be referred back to the user. Under no circumstances will new bids be placed where the amount is

greater than the maximum price limit P_{MAX}. If all bids are placed, this will result in j_i bids of amount $b^i_{mi-ji} + \delta_i$ being placed in each auction A_i.

Referring to Fig. 15 herein, there is illustrated schematically an offer selection algorithm for selecting a set of offers to sell a plurality of lots of goods or services in a plurality of various auctions purchasing goods or services from the trading device. The trading device attempts to sell goods by making offers to a competitive tender. Hence the 'auctioneer' is putting out goods to competitive tender. The auctioneer will never receive bids, only offers.

The offer selection algorithm illustrated in Fig. 15 operates where a plurality N of auction entities request goods or services sold by the trading device 400 which offers those goods and/or services for sale. Each individual auction i buys a quantity M_i lots of goods or services. That is to say each auction entity i posts on its website details of the quantity of lots of goods or services which it wishes to acquire, and posts a call for offers for each of those lots of goods or services.

The trading device 400 places a plurality of offers with a selection of a plurality of N auction entities. The number of individual offers, the particular auction with which the offer is placed, and the amount of the offer are determined by the offer selection algorithm illustrated schematically in Fig. 15 herein.

The trading device monitors the plurality N auctions, each auction entity wishing to acquire a quantity M_i goods, the number M_i varying from auction to auction in the general case.

In step 1500, the user inputs details defining the quantity of goods or services M required to be sold or supplied into the trading device 400 using the user interface. These details are stored in local memory in the trading device for

further reference. In step 1501, the user inputs an absolute price limit P_{MIN} for each lot of goods or services which are to be sold, the absolute price limit P_{MIN} being the price below which the user is unwilling to sell or supply the goods or services. A separate price limit P_{MIN} is input for each lot of goods or services to be sold or supplied. The data describing the minimum offer price P_{MIN} is stored in memory by the trading device. In step 1502, the user inputs a referral price data P_{REF}, for each lot of goods or services which are to be sold or supplied. The referral price is the price below which the trading device cannot operate automatically, but must refer all offers which it makes to the plurality N of auctions back to the user via the user interface, before proceeding to place those offers. A user behaving optimally will set the minimum price P_{MIN} to be the same as the referral price P_{REF}, so that in practice the trading device once instructed to sell items does not refer back to the user. In step 1503, the trading device monitors the plurality of N auctions, by parsing data from the websites operated by those auction entities, the data describing the quantity, and offer price for goods or services of the type specified in step 1500. Auction entities may be selected on the basis of reliability, settlement terms, delivery terms, and trading history. In the best mode implementation, a user will personally select the plurality of N auction entities from which the price calculation algorithm can choose, rather than leaving the selection process to a further set of auction selection algorithms 702.

In step 1504, the algorithm monitors a plurality of offers from each of the plurality N auctions, receiving the lowest current offers in each auction i of the plurality. Where a large number of offers are posted by an auction entity, the algorithm may be configured to only receive from each auction at most a number of lowest offers S_1^i, ... S_{mi}^i being the same as the quantity Q of the lots of goods/services to be disposed of. This prevents the trading device becoming flooded with large quantities of offer data from the plurality of auction entities. Auction entities may be pre-selected by the user in a similar fashion as described herein above with respect to an operation by the trading device.

In step 1504 for each auction i the lowest M_i offers S_1^i, ... S_{mi}^i are stored. In step 1505, for each auction, the current number of active placed offers which the trading device has placed in the auction are stored. In step 1506, for all selected auctions, the number of active offers of goods/services specified are summed. In step 1507, if the number of current active offers is greater than M, then the algorithm returns to step 1503 and the monitoring process is continued. However, in step 1507, if the number of current active offers is less than M, then in step 1508 the algorithm proceeds to place a set of new offers according to the offer selection algorithm.

Selection of individual auctions with which to trade are chosen from the selected plurality N of auctions, and the prices at which those trades occur is carried out by the offer selection algorithm illustrated with reference to Fig. 16 herein. The basis for operation of the algorithm of Fig. 16 is as follows.

In step 1601, the plurality N of pre-selected auctions are monitored by parsing data from the websites of those auction entities. Offer data S_1^i...., S_N^i, for a plurality of individual offers S for each auction A_i up to the maximum number of auctions N is received by the trading device, suitably in HTML, or XML languages.

For an auction A_i, a beatable – j set of offers is defined as being:

$$(S_{mi-j+1}^i, \ldots\ldots, S_{mi}^i) \text{ for } j \geq 1.$$

The beatable – 0 set for any auction is defined to be the empty set.

In step 1602, there are generated all sets of offers which can be formed, by taking the members of exactly one beatable – j set ($j \geq 0$) from every auction A_i. Let Φ be the set of all such sets.

All sets in Φ which do not contain a number exactly Q of offers made by parties other than the trading device are deleted or ignored in step 1603. The remaining offer sets each having Q members are selected in step 1604.

For each set, $D \in \Phi$ a profit D is defined as

Equation 6

$$\text{Profit}(D) = \sum_{i=1}^{N} j_i (S_{mi-ji+1} + \delta) - \sum_{S \in D_A} S$$

Where D_A is the set of offers in set D made by the trading device.

In step 1605, a maximum offer set D_{MAX} is selected by choosing $D_{MAX} \in \Phi$ such that

$\text{Profit}(D_{MAX}) \geq \text{profit}(D)$
for all $D \in \Phi$.

In step 1606 offers are deployed in the plurality of auctions by placing j_i offers each of amount

$S_{mi+j-1}{}^{i} - \delta$

in each auction A_i, where δ is an offer increment selected by the user or specified by the auction site.

Once the offers are placed with the selected auctions, then provided no competing offers are made by third parties at a lower price than those made by the trading device, then the offers should be accepted by the selected auctions.

Different numbers of lots of goods or services may be accepted by different auction entities, possibly at different times. Thus, the algorithm of Fig. 16 may dynamically and continuously loop back to step 1601 further monitoring the plurality of N auctions, reviewing any offers which have not yet been accepted. If an offer is accepted, then for that particular offer which is accepted, settlement occurs in step 1608, by the algorithm confirming by a message to the auction entity concerned that the offer has been accepted. This message forms a legally binding contract between the trading device and the auction entity. Payment and delivery may occur by conventional means, for example physically shipping the goods, and making a credit card transaction, or electronic funds transfer by conventional means. In the best mode implementation described herein, settlement of trades may be delegated to a human user for actual implementation.

Appendix 1

Algorithm design for Agents which participate in multiple simultaneous auctions

Chris Preist
Hewlett-Packard labs
Filton Rd
Stoke Gifford
Bristol BS34 8QZ
UK
Tel: +44 117 312 8311
Fax: +44 117 312 9250
Email: cwp@hplb.hpl.hp.com

Abstract:
In this paper, we discuss the design of algorithms for agents to use when participating in multiple simultaneous English auctions, aiming to purchase multiple goods. Firstly, we present a coordination algorithm, which ensures the agent places appropriate bids in the different auctions to buy exactly the right number of goods. Secondly, we combine this with an algorithm to determine what maximum bid an agent should place in an auction which is about to terminate. This algorithm combines a belief-based model of the auctions with a utility analysis. This analysis is to trade off the certain outcome of the terminating auction against the possible outcomes of the remaining auctions, and hence place appropriate bids in each.

1. Introduction

Electronic commerce [1] is having a revolutionary effect on business. It is changing the way businesses interact with consumers, as well as the way they interact with each other. Electronic interactions are increasing the efficiency of purchasing, and are allowing increased reach across a global market.

E-commerce is not a static field, but is constantly evolving to discover new and more effective ways of supporting business. Initially, e-commerce involved the use of EDI and intranets to set up long-term relationships between suppliers and purchasers. This increased the efficiency and speed of purchasing, but resulted in lock-in in the relationship. Both suppliers and purchasers had to invest significantly up-front in the relationship, so were not easily able to move their business elsewhere. The technological relationship between the parties was a friction factor, preventing free competition in the longer term [2]. Often, the relationship was (and still is) beneficial to both parties. However, the lock-in effect also meant that when the relationship became less beneficial to one party, they couldn't easily move elsewhere.

The second phase of e-commerce aimed to address this problem. With the increasing availability of the web, a more open e-commerce environment is developing, allowing businesses to trade more flexibly with each other. Some of this openness is achieved by competition between web portals, while some competition occurs within a single web portal, acting as a marketplace for buyers and sellers to meet. Some of the efficiencies of EDI can now be achieved in a more open environment, where relationships no longer need to be long-term.

However, there is a benefit of the EDI approach that is often lost in this new phase. Price negotiation was carried out in advance in the EDI world, so purchasing can be entirely automated. When a manufacturing planning and forecast system identifies the need for a purchase, it can initiate it automatically without any human involvement, increasing speed and efficiency. In phase two, each purchase may involve interaction with a new supplier, and so may involve new negotiation of terms. As a result of this, many of these purchases can't be made automatically, and instead require human interaction, mediated by the web.

The third phase of electronic commerce is just beginning. It aims to address this issue, allowing automated business interactions to take place in a fluid environment. Technology will no longer be a friction factor to changing supplier or customer. Long-term relationships will still play an important role, but they will persist because of the choice of both parties rather than technological lock-in. The key building blocks of this new world, e-services, will be able to interact dynamically with each other to create short-term or long-term trusted trading relationships to satisfy the needs of different business partners. Many technologies are involved in this development – distributed systems, encryption and PKI, XML and associated business ontologies, economic analysis and game theory, to name just a few. As automation and distribution are central to the vision, agent technology provides a fundamental role in this.

In Section 2, we discuss the role of auctions in electronic commerce, present the problem of multiple auction participation and give an overview of tools used to support participation in auctions currently. In section 3, we present a basic coordination algorithm used to bid effectively in multiple auctions which terminate simultaneously. In section 4, we extend this algorithm by providing a belief-based learning component, and use utility analysis to determine whether it is worth making a purchase in an auction about to terminate. In section 5, we present related work, and in section 6 we conclude with future directions for this work.

2. Auctions in Electronic Commerce

In phase two of the e-commerce revolution, auctions have become increasingly important both for business to business transactions and for consumer purchasing. Many different auction designs are possible. Game theory [3] can be applied to study how best to design them for specific situations [4]. Among the most popular designs are English, Dutch and Vickrey auctions. In an English auction, a seller offers a good for sale and buyers bid the price they are willing to pay. Each bid announced must be greater than the previous bid, and the item is sold to the highest bidder. In a Dutch auction, the process runs in reverse. The seller

announces a proposed price, and buyers can accept it. As time progresses, the seller decreases the proposed price until a buyer accepts. In a Vickrey auction, bidders place their bids in a sealed envelope and submit them to a trusted third party. At a certain time, the party opens the envelopes, and the good is sold to the highest bidder at the second highest price. Under certain conditions, these three auction formats can be shown to produce an identical revenue for the seller.

As a result of this explosion in popularity, more and more companies are offering auction sites. Because of this, if you want to purchase a particular good, there are often many auction sites which are offering it. If you really want to get the best price, you must monitor all of these auctions using your web browser, and place bids appropriately. Care must be taken, to ensure you don't make more than one purchase! If there are a large number of auctions, this can be quite a daunting task, requiring your undivided attention for a period of time. Furthermore, if you wish to purchase more than one item, (as is often the case in B to B trading,) it becomes almost impossible.

As a result of this, auctions are beginning to offer support tools to make your job easier. Search tools such as Auction Beagle (http://www.auctionbeagle.com/) and Auction Octopus (http://www.auctionoctopus.com/) allow you to locate and monitor auctions selling specific goods, thus eliminating the need to have 20 browsers open at once. Auction Rover (http://www.auctionrover.com/) provides price trend information on popular items, to aid you in deciding what maximum price to bid. These help, but still leave a large amount of work for you to do. Particularly in the B-to-B context, this can result in an unacceptable overhead. To eliminate this, it is necessary to design automated agents able to carry out the task on your behalf.

Some sites running English auctions, such as Auction Sales (http://www.auction-sales.com/), do offer a simple bidding agent. This agent resides on the auction site, and bids on your behalf. You enter the maximum you are willing to pay, and it places the lowest possible bid on the web site (Either the reservation price, or if bidding has already started, it bids just above the current highest bid.) If all bidders in an auction use such an agent, the auction becomes a Vickrey style auction instead of an English auction. (The sale is made at second price plus the minimum bid increment.)

However, such agents are not able to participate in multiple auctions, either on the same web site or across different ones. As a result, once you use the agent, you are committed to making a purchase on the site if you can. Locally, you may pay the lowest price you can to win the auction. However, from a global perspective, that particular auction is unlikely to have been the best place to make a purchase.

These bidding agents have an additional disadvantage. To use them, you must reveal the highest price you are willing to pay to the auction site. This gives the auction site information that could be used to cheat you. Furthermore, as auction sites often receive percentage commission on sales made, they also have an incentive to cheat you. To do this, they would take note of the highest price you are willing to pay, and enter a mythical bid in the auction (a 'shill') just under this price. Your agent would then place your maximum bid, and the seller has made a sale to you at the best possible price (assuming you are the highest bidder.) The auctioneer would therefore collect their maximum possible commission.[1]

Hence, neither the auction search facilities nor the existing bidding agents provided by current technology are adequate to meet the needs of a B-to-B auction trader. In this paper, we propose a solution to this problem. We present algorithms that can be used by an agent to participate in multiple auctions on behalf of a trader, and can lead to optimal or near-optimal purchase decisions being made.

The agent aims to purchase one or more identical goods on behalf of its user. It can participate in many auctions for this good, and coordinates bids across them to hold the lowest bids. As auctions progress, and it is outbid, it may bid in the same auction or choose to place a bid in a different auction. The algorithm consists of two parts. Firstly, it has a coordination component, which ensures it has the lowest leading bids possible to purchase the appropriate number of goods. Secondly, it has a belief-based learning and utility

[1] While this is possible, I make no claims that it actually happens.

analysis component, to determine if it should deliberately 'lose' an auction in the hope of doing better in another auction later.

3. The Coordination Algorithm

The basic algorithm uses the coordination aspect only. If all auctions terminate simultaneously, it will place optimal bids, and make the required purchase at the lowest price. However, if some auctions terminate later than others, and new auctions may come into being, then its behaviour is non-optimal, and needs to be supported by belief-based and economic techniques, to be discussed in the next section.

Definitions

An *auction house* may run one or more *auctions* for a given good. Each auction a_i offers $n(a_i)$ goods for sale. Auctions are assumed to be English auctions in format, with *bidders* bidding the price they are willing to pay for the good. A bidder may place more than one bid in a given auction. The $n(a_i)$ goods offered in the auction are sold to the bidders making the $n(a_i)$ highest bids, for the price they bid. In case of two equal bids, the item goes to the earliest bidder. Hence the auction is *discriminatory* – some buyers will pay more than others for the same good. Different auctions impose different rules covering how a bid may be entered or retracted. For the purposes of this paper, we assume that a buyer may not retract a bid, and a buyer may enter a bid provided it is not lower than $n(a_i)$ existing bids.

We will define the algorithm, and simultaneously present an example to illustrate the definitions. The illustrative paragraphs, provided for clarification, are indented.

Our agent participates in many auctions selling similar goods, spread out between many auction houses. It wishes to purchase m goods in these auctions, and is given a valuation of v on each good by its user. To do this, it monitors the set of auctions currently progressing. For each auction a_i, it observes the n_i highest bids. In other words, it observes the values of the bids which, if the auction terminated immediately, would result in a successful purchase. Let these bids be labelled $\{b_1^i,..,b_{ni}^i\}$, where b_1^i is the highest bid in the auction, and b_{ni}^i is the lowest bid which would currently succeed. We refer to these as the currently *active* bids. To represent the reservation price r, we assume that the seller places n_i bids of value r-δ, where δ is the minimum bid increment.

> Let us assume our agent is attempting to purchase 6 discount PCs of a given specification. There are 3 auction houses, each running one auction to sell PCs of this specification. Auction a_1 is selling 4 PCs, auction a_2 is selling 3 PCs and auction a_3 is selling 2 PCs. Assume all auctions have a minimum bid increment of 1.
>
> Auction a_1 currently has the following bids registered (Starred bids are by our agent);
>
> 180   180   160**   150   110**   110   100   100
>
> As the auction is for 4 items, the agent observes the 4 highest bids, $\{b_1^1=180, b_2^1 = 180, b_3^1 = 160, b_4^1 = 150\}$
>
> Auction a_2 has the following bids registered;
>
> 160**   155   150   100   100
>
> The agent observes the bids $\{b_1^2=160, b_2^2 = 150, b_3^2 = 150\}$
>
> Finally, auction a_3 has just opened, and has no bids. It has a reservation price of 100. The agent represents this by observing the fictitious bids $\{b_1^3=99, b_2^3 = 99\}$ by the seller.
>
> Our agent holds 2 active bids, and so needs to place bids to gain an additional 4.

Let L be the number of currently active bids that are held by our agent. (Initially, L will be zero.) To ensure it makes m purchases, it needs to make new bids that result in it having an additional (m-L) active bids. As we shall see, this may require it to make more than (m-L) bids, as it may need to outbid itself.
If the agent is to hold j active bids in auction a_i, it must place bids that beat the lowest j of the currently active bids. We define the *beatable-j list* for auction a_i to be the ordered set of these bids – namely bids $\{b_{ni-j+1}^i,..,b_{ni}^i\}$. To beat the bids in this list, the agent must place j bids of value $b_{ni-j+1}^i+\delta$ where δ is the minimum bidding increment. The *incremental cost* to the agent of placing these bids, if successful, above the cost that it would have incurred in auction a_i previously, is $j*b_{ni-j+1}^i+\delta$ - {sum of previous bids in a_i}. The beatable-0 list of any auction is defined to be the empty set, and has incremental cost of zero. Obviously, an auction for q goods has no beatable-j lists for $j>q$.

> In the above example, the beatable-1 list of auction a_1 is {150}, with incremental cost 151. The beatable-2 list is {160, 150}, with incremental cost 162 (because the agent already holds the bid at 160). Similarly the beatable-3 list is {180, 160, 150} with incremental cost 383, and the beatable-4 list is {180, 180, 160, 150} with an incremental cost of 564.

The agent now constructs potential *bid sets*. A bid set is a set of beatable-j lists that satisfies the following criteria;

1. The set contains exactly one beatable-j list from each auction. (In the terminology of Reiter and the model-based diagnosis community, the set constructed is a *hitting set* of the sets of all beatable-j lists associated with each auction.)
2. The beatable-j lists contain, in total, exactly (m-L) bids made by parties other than our agent.

In other words, each bid set represents one possible way of placing bids to ensure that our agent will gain an additional (m-L) active bids, and therefore will hold exactly m active bids. We define the incremental cost of each of these bid sets to be the sum of the incremental costs of the beatable-j lists in it.

> Returning to our example, our agent needs to find bid sets containing exactly 4 bids made by parties other than it. An example bid set satisfying the above criteria would be;
> [{180, 160,150}, {150}, {99}]
> This set is made from the beatable-3 list of auction a_1, and the beatable-1 lists of auctions a_2 and a_3. Its incremental cost is 634.

The agent must generate the bid set with the lowest incremental cost. In addition, it must avoid generating bid sets that contain a bid equal to or greater than its valuation of the good, v. Various algorithms can be used to do this. The simplest is to generate all possible bid sets, filter out those containing bids greater than v, and select the one with lowest cost. However, this is clearly inefficient, and alternative more intelligent search techniques would be better. Discussion of such algorithms is beyond the scope of this paper.

If there is more than one bid set with identically lowest cost, the agent chooses one arbitrarily. If no such bid sets exist, the agent relaxes condition 2 and finds the smallest i such that at least one bid set exists which contains (m-L-i) bids made by parties other than the agent. Given this i, the agent chooses the bid set with the lowest incremental cost.

Having generated the bid set with the lowest cost, the agent places bids in each auction. For each beatable-j list {bni-j+1I,..,bniI} in the bid set, the agent places j bids of value bni-j+1I + d in the corresponding auction a_i.

> In our example, the bid set with lowest cost is [{150} , {150}, {99,99}]
> This set has cost 502. The agent therefore places one bid of 151 in each of auctions a_1 and a_2, and two bids of 100 in auction a_3, and thereby gains 6 active bids.

The agent continues to monitor the auction, and repeats its analysis if other parties place new bids. In this way, the agent ensures it maintains m active bids at the least possible cost to itself, unless doing so requires it to place bids above its valuation of the good. Providing all auctions terminate simultaneously, this will result in it buying the goods at the best price possible, given the competition in each auction.

4. Dealing with Different Auction Deadlines

Now, we consider the case where auctions terminate at different times. In such a situation, the algorithm above will not necessarily behave optimally. Imagine a situation where an auction starts every half hour, and lasts for an hour. The agent would always monitor two auctions, one which is nearer closing than the other. Inevitably, bids will be higher in the auction which is nearing completion. Hence the agent would switch bidding to the newer auction, and withdraw from the auction about to close. If this continued, the agent would never make a purchase, but would simply switch bids to a new auction every half hour.

The agent needs a mechanism for determining whether to remain in an auction which is about to close, even when there are other auctions with lower current bid prices. To do this, it must be able to make a trade-off in terms of expected value between the relative certainty of remaining in an auction about to close, against the risk of participating in a newer auction. The newer auction may result in a lower purchase, or may result in a far higher purchase price above the agent's valuation of the good. In this section, we propose a mechanism for doing this.

The mechanism we use combines belief-based learning with utility theory. The agent uses belief-based learning to build a model of the spread of valuations held by participants in different auction houses. Then, based on its beliefs about these valuations, it calculates the utility of likely participation in persisting auctions, and compares this with the certain outcome in the terminating auction. If the terminating auction has a higher utility, it remains a participant and makes the purchase. If the remaining auctions have higher expected utility, it withdraws from the terminating auction and continues participation elsewhere.

4.1 The belief-based learning mechanism

The agent generates a model of the potential outcome of auctions by creating a model of each auction house. For a given auction house and a given type of good, it creates a belief function $B(x,y)$ representing the probability that x bidders value the good with a valuation greater than y in a given auction for that good. It builds up this function by monitoring auctions for the good conducted by the auction house. Various possible belief learning techniques can be used to generate this function. For the purposes of this paper, we present a simple example based on that used in [21].

The belief function is defined iteratively – The initial beliefs after one auction $B_1(x,y)$ are defined, and the beliefs after the t+1th auction $B_{t+1}(x,y)$ are defined in terms of the beliefs $B_t(x,y)$ held prior to the auction.

$B_1(x,y) = 1$ if x or more bidders have placed a bid of y or greater in the first auction observed,
0 otherwise.

$B_{t+1}(x,y) = ((t\ B_t(x,y)\ +1)\ /\ (t + 1)$ if x or more bidders have placed a bid of y or greater in the t+1th auction observed,
$= t\ B_t(x,y)\ \ /\ (t + 1)$ otherwise

More sophisticated versions, for example giving more weight to recent auctions or using the statistical switching technique of [21], may be appropriate depending on the learning environment. However, discussion of these is beyond the scope of this paper.

Using this function, we can estimate the probability that a bid of a certain value will be successful in an auction by a given auction house. Consider an auction for n goods, in which our agent wishes to purchase one. The probability that a bid of x by our agent will be successful can be estimated to be $1 - B(n,x)$; i.e. 1 minus the probability that n other bidders are prepared to outbid our agent.

There is a flaw in this model, which must be taken into account if it is to be successful. Unlike a Vickrey or Dutch auction, an English auction reveals nothing about the valuations of successful bidders. In other words, if a bidder makes a successful bid of x, we cannot be sure how much higher they may have been

willing to bid. To take account of this, it is necessary to add some kind of heuristic weighting to the belief function – we must increase the value of a successful bid by a certain amount, to reflect this possible willingness to bid higher. One possibility is to add a small random amount to each successful bid. In some domains, it may be possible to use econometric data to determine accurately the range that this should be drawn over, while in other domains it may be necessary to use a heuristic estimate.

4.2 Utility analysis of leaving an auction

We now consider how this belief function can be used to compare the expected payoff of an auction which is about to terminate with the less certain outcome of other auctions which terminate later. For the sake of clarity and brevity, we present the technique assuming our agent wishes to purchase a single good.

The expected payoff from the terminating auction is simple to calculate. Assuming our agent is holding an active bid x, or is able to place one at the last moment, then the payoff will be (v-x). If the agent is unable to place a bid because all active bids are beyond its valuation of the good, then payoff will be zero and the agent is forced to participate in other auctions.

Again, we will introduce an example to illustrate the principles being presented. Assume our agent is purchasing one good from one of three auctions. Auction a_1 is nearing completion, while auctions a_2 and a_3 are continuing. Each auction is for 2 goods, and are run by separate auction houses. The active bids are as follows;

Auction a_1:	140	130
Auction a_2:	115	110
Auction a_3:	100	95

Our agent values the good at 200, so could continue bidding in auction a_1. However, should it, or should it switch to the other auctions where prices are lower?

The expected payoff of continuing to participate in the non-terminating auctions is more complex to calculate. To do this, we use the belief function to calculate the probability our agent will be able to make a purchase at various possible bid prices. Recall that, for a given bid price x, the probability our agent will make a successful bid in an auction run by a given auction house is 1-B(n,x), where n is the number of goods being sold. Similarly, the probability that our agent will be able to make a successful bid at a lower price, x-1, is 1-B(n,x-1). Hence, the probability that our agent will succeed with a bid of x and no lower is B(n,x-1)-B(n,x). The utility of this outcome will be (v-x). Hence, we can calculate the expected utility of participating in a given auction as;

$$\sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x)$$

Of course, as the auction may already be in progress, it is necessary to take into account the current active bids in that auction. The general belief function for the auction house is therefore adapted for this particular auction by setting B(n,x) = 1 if there are n active bids at or over x.

To return to our example, let us assume that our agent has built up the following belief function for the auction house running auction a_2;

Bid price x:	105	110	115	120	125	130	135	140	145
B(2,x):	1	0.8	0.7	0.6	0.6	0.5	0.3	0.3	0

As there are already bids of 110 and 115, B(2,110) becomes set to 1 for this auction.

As our agent has a valuation of 200, we can calculate the expected utility of this auction to be;

$$\sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x) = (0.3*85)+(0.1*80)+(0.1*70)+(0.2*65)+(0.3*55) = 63.7$$

Given an expected utility on the remaining auctions, the agent must decide whether to place higher bids in the auction which is about to terminate, or withdraw from it. If we assume that the agent is risk neutral, then it will be willing to bid up to a value where the actual utility of the terminating auction is the same as the highest expected utility among the remaining auctions. In other words, it is prepared to make a maximum bid b_{max} of;

$$b_{max} = v - \sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x)$$

In our example, assuming that a_3 has a lower expected utility than a_2, our agent will be willing to bid up to (200-63.7) = 136.3 in auction a_1. Hence, it will place a bid of 135, and will withdraw if this is outbid. In this case, it will hope to make a better purchase in auctions a_2 or a_3.

In this way, the agent is able to make informed decisions about whether to continue bidding in an auction or to switch. If it is making multiple purchases, it may purchase some in the terminating auction, and choose to switch others to continuing auctions. Extensions of the algorithm to handle this case will be dealt with in a future paper.

5. Related Work

Research into automated negotiation has long been an important part of distributed AI and multi-agent systems. Initially it focused primarily on negotiation in collaborative problem solving, as a means towards improving coordination of multiple agents working together on a common task. Laasri, Lassri, Lander and Lesser [5] provide an overview of the pioneering work in this area. As electronic commerce became increasingly important, the work expanded to encompass situations with agents representing individuals or businesses with potentially conflicting interests. The contract net [6] provides an early architecture for the distribution of contracts and subcontracts to suppliers. It uses a form of distributed request-for-proposals. However, it does not discuss algorithms for determining what price to ask in a proposal. Jennings et.al. [7] use a more sophisticated negotiation protocol to allow the subcontracting of aspects of a business process to third parties. This is primarily treated as a one-to-one negotiation problem, and various heuristic algorithms for negotiation in this context are discussed in [8]. Vulkan and Jennings [9] recast the problem as a one-to-many negotiation, and provide an appropriate negotiation protocol to handle this. Other relevant work in one-to-one negotiation includes the game-theoretic approach of [10] and the logic-based argumentation approach of [11].

As much electronic commerce involves one-to-many or many-to-many negotiation, the work in the agent community has broadened to explore these cases too. The Michigan AuctionBot [12] provides an automated auction house for experimentation with bidding algorithms. The Spanish Fishmarket [13] provides a sophisticated platform and problem specifications for comparison of different bidding strategies in a Dutch auction, where a variety of lots are offered sequentially. The Kasbah system [14] featured agents involved in many-to-many negotiations to make purchases on behalf of their users. However, the algorithm used by the agents (a simple version of those in [8]) was more appropriate in one-to-one negotiation, and so gave rise to some counter-intuitive behaviours by the agents. [15] and [16] present adaptive agents able to effectively bid in many-to-many marketplaces. [17] demonstrates how these can be used to produce a market mechanism with desirable properties. Park et.al. [18][19] present a stochastic-based algorithm for use in the University of Michigan Digital Library, another many-to-many market.

Gjerstad et. al. [20] use a belief-based modeling approach to generating appropriate bids in a double auction. Their work is close in spirit to ours, in that it combines belief-based learning of individual agents bidding strategies with utility analysis. However, it is applied to a single double auction marketplace, and does not allow agents to bid in a variety of auctions. Vulkan et.al. [21] use a more sophisticated learning mechanism which combines belief-based learning with reinforcement learning. Again, the context for this is a single double auction marketplace. Unlike Gjerstad's approach, this focuses on learning the distribution of the equilibrium price. Finally, the work of Garcia et.al. [22] is clearly relevant. They consider the development of bidding strategies in the context of the Spanish fishmarket tournament. Agents compete in a sequence of Dutch auctions, and use a combination of utility modeling and fuzzy heuristics to generate

their bidding strategy. Their work focuses on Dutch rather than English auctions, and on a sequence of auctions run by a single auction house rather than parallel auctions run by multiple auction houses. However, the insights they have developed may be applicable in our domain also. We hope to investigate this further in the future.

6. Conclusions and Future Work

By interleaving the application of two algorithms of the form described above, our agent can effectively participate in multiple English auctions. It will use the coordination algorithm to place lowest possible bids across auctions. It will use the bid withdrawal algorithm to determine when it is worth bidding higher in an auction which is about to terminate, as opposed to transferring to other auctions where the active bids are currently lower.

In this paper, we have sketched out the structure of appropriate algorithms to do this. The specifics of these algorithms, particularly the bid withdrawal mechanism, may need refinement and specialisation to operate in specific market applications. Furthermore, the richness of the model may be increased. Specifically;

- A more sophisticated belief mechanism could be used. This may be generic, or could be tailored to the specific dynamics of a particular market.
- The algorithm presented assumes that the buyer is risk-neutral. This should be generalised to allow the agent to adopt other risk attitudes as appropriate.
- The algorithm does not take into account time discounting or deadlines. Again, this should be generalised.
- The algorithm assumes that the agent values all goods equally. This should be generalised to allow the agent to receive a demand curve from its user.
- The algorithm assumes that the beliefs about auctions are accurate. By measuring the deviation of actual auctions from the predictions, it would be possible to give a measure of confidence in the belief. This could be used to moderate the agent's decision to switch auctions, taking into account the agent's attitude to risk.
- The algorithm only takes into account the expected payoff of existing auctions. It may be appropriate to also model the possibility that an auction house may bring new auctions into being, and the potential payoff of such auctions.
- Throughout this paper, we have assumed that all auctions are English in format. Research is required to generalise this to cover other forms of auction, such as Dutch, Vickrey and Double auctions.

We hope to address some of these issues in a future paper.

References

[1] R. Kalakota and A. Whinston. Frontiers of Electronic Commerce.. Addison-Wesley 1996.
[2] C. Shapiro and H. Varian. Information Rules – A Strategic Guide to the Network Economy. Harvard business school press, 1999.
[3] K. Binmore Fun and Games.. D. Heath and Co. 1992.
[4] Paul Klemperer, "Auction theory: a guide to the literature" J. of Econ. Surveys, Vol. 13, No. 3, July 1999, pp. 227-286
[5] B.Laasri, H.Laasri, S. Lander and V. Lesser. A Generic Model for Intelligent Negotiating Agents. International Journal of Intelligent and Cooperative Information Systems, 1(2) 1992 pp291-317.
[6] R.G. Smith . The contract net protocol: high-level communication and control in a distributed problem solver.. *IEEE Trans. Comput.*, **29**, 1104-1113, 1980.
[7] N.R. Jennings, P. Faratin, M.J. Johnson, P.O. O'Brien and M.E. Wiegand. Using Intelligent Agents to Manage Business Processes. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM-96)*, 345-360, April 1996
[8] P. Faratin, C. Sierra and N. Jennings. Negotiation decision functions for autonomous agents. Robotics and Autonomous Systems 24(3-4),1998, pp159-182.

[9] N.Vulkan and N.Jennings. Efficient Mechanisms for the Supply of Services in Multi-Agent Environments, Proceedings of the 1st International Conference on the Internet, Computing and Economics, ACM Press 1998.

[10] J. Rosenschein and G. Zlotkin. Rules of Encounter. MIT Press, 1994.
[11] S. Parsons, C.Sierra and N.Jennings. Agents that reason and negotiate by arguing.
[12] P.Wurman, M.Wellman and W.Walsh. The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents. In Proc. Second Conference on Autonomous Agents, 1998.
[13] J. Rodriquez-Aguilar, P. Noriega, C. Sierra and J.Padget. Fm96.5: A Java-based e,ectronic auction house. Proc. Second International Conference on the Practical Application if Intelligent Agents and Multi-Agent Systems, 1997, pp207-224.
[14] A. Chavez, D.Dreilinger, R.Guttman and P.Maes. A Real-Life Experiment in creating an Agent Marketplace. Proc. Second International Conference on the Practical Application if Intelligent Agents and Multi-Agent Systems, 1997.
[15] Cliff, D. and Bruten, J. Less than Human: Simple adaptive trading agents for CDA markets. Proceedings of the 1998 Symposium on Computation in Economics, Finance, and Engineering: Economic Systems
[16] Preist, C. and van Tol, M. Adaptive Agents in a Persistent Shout Double Auction. Proceedings of the 1st International Conference on the Internet, Computing and Economics, ACM Press 1998.
[17] C.Preist. Commodity Trading using an Agent-Based iterated Double Auction. Proc. Third Conference on Autonomous Agents, 1999.
[18] S. Park, E.Durfee and W.Birmingham. Emergent Properties of a Market-Based Digital Library with Strategic Agents. Proc. International Conference on Multi Agent Systems, 1998.
[19] S. Park, E.Durfee and W.Birmingham. An Adaptive Agent Bidding Strategy based on Stochastic Modelling. Proc. Third Conference on Autonomous Agents, 1999.
[20] Gjerstad, S. and Dickhaut, J. Price formation in double auctions. Games and Economic Behaviour, 22(1), pp1-29, 1998.

[21] N. Vulkan and C. Preist. Automated Trading in Agents-based Markets for Communication Bandwidth. Proc. UKMAS 1999.
[22] P. Garcia, E.Giminez, L. Godo and J. Rodriguez-Aguilar. Possibilistic-based design of bidding strategies in Electronic Auctions. Proc. 13th Biennial European Conference on Artificial Intelligence, 1998.

Claims:

1. An electronic trading entity comprising:

a processor;

a memory means;

a communications means;

monitoring means for monitoring a plurality of trading data displayed by a plurality of auction entities;

quantity calculation means for calculating quantities of items for trading with said plurality of monitored auction entities,

price calculation means for calculating optimal prices of items to be traded with said plurality of auction entities;

wherein said price calculation means operates to process said trading data to obtain trading prices for a quantity of items calculated by said quantity calculation means, said prices calculated for a set of chosen auction entities selected from said set of all said plurality of auction entities;

said price calculation means operating to process said selected data to obtain trading price for a predetermined amount of items across all said selected set of said plurality of auction entities.

2. The electronic trading entity as claimed in claim 1, further comprising search means for searching electronically for individual ones of said auction entities.

3. The trading entity as claimed in claim 1, wherein said amount calculation means comprises:

means for storing bid data describing a plurality of bids in at least one of said plurality of auction entities;

means for storing user bid data describing a number of bids placed by said trading entity with at least one of said plurality of auction entities; and

means for determining an optimum number of bids and corresponding price amounts of said bids to place with at least one of said plurality of auction entities.

4. The trading entity as claimed in claim 1, wherein said quantity calculation means comprises:

means for storing an offer data describing a plurality of offers made by said plurality of auction entities;

means for storing user offer data describing a number of offers placed by said trading entity with at least one of said plurality of auction entities; and

means for determining an optimum number of offers and corresponding price amounts of said offers to place with at least one of said plurality of auction entities.

5. A method of operating an electronic trading entity for trading a plurality of tradable items, said method comprising the steps of:

monitoring a plurality of auction entities;

inputting a plurality of trading data from said plurality of auction entities;

processing said trading data of said plurality of auction entities to determine an optimum set of amount data describing quantities and prices of said tradable items for trading by said trading entity.

6. The method as claimed in claim 5, wherein said step of processing trading data of said plurality of auction entities comprises:

storing in memory a plurality of bid data displayed by said plurality of auction entities;

determining a number of active bids of said trading entity already sent by said trading entity to at least one of said plurality of auction entities;

constructing a plurality of currently successful sets of said bid data of said plurality of auction entities;

determining a cost to outbid each said set of currently successful bids;

selecting a said currently successful set of bids having an optimum cost to outbid; and

constructing a set of bids of said trading entity which minimally outbid said optimum cost to outbid currently successful bid set.

7. The method as claimed in claim 5, wherein said step of constructing a set of bids of said trading device which are higher than said optimum cost currently successful bid set comprises:

for each bid of said optimum bid set, said bid made either by said trading entity or a third party, selecting a highest third party bid appearing in an identical auction entity to said bid, which also appears in said optimum bid set; and

generating a trading device bid of an amount equal to said highest third party bid plus a minimum bid increment.

8. The method as claimed in claim 5, wherein said step of processing trading data of said plurality of auction entities comprises:

storing in memory a plurality of offer data of said plurality of auction entities;

determining a number of active offers of said trading entity already sent by said trading entity to at least one of said plurality of auction entities;

constructing a plurality of currently successful sets of said offer data of said plurality of auction entities;

determining a value to undercut each said set of currently successful offers;

selecting a said currently successful set of offers having an optimum value to undercut; and

constructing a set of offers of said trading device which minimally undercut said optimum value to undercut currently successful offer set.

9. The method as claimed in claim 5, wherein said step of constructing a set of offers of said trading device which are lower than said optimum value to undercut currently successful offer set comprises;

for each offer of said optimum offer set, said offer made either by said trading entity or a third party, selecting a lowest third party offer appearing in an identical auction entity to said offer, which also appears in said optimum offer set; and

generating a trading device offer of an amount equal to said highest third party offer minus a minimum bid decrement.

10. A method of exchanging data between a first electronic trading entity and a plurality of electronic auction entities, said method comprising the steps of:

said first trading entity monitoring data displayed by said plurality of auction entities;

said first trading entity extracting from said monitored data, data describing prices of individual items to be traded;

said first trading entity determining from said price data an optimum set of trading data for sending to said plurality of auction entities; and

said first trading entity communicating said trading data to said plurality of auction entities.

Abstract

AUCTION METHOD AND APPARATUS FOR ELECTRONIC COMMERCE

An electronic trading entity (400) comprises a computing device having a processor, memory, user interface, and communications functionality, the computer entity comprising an algorithm (700) for monitoring a plurality of auction entities remotely over the internet; a algorithm (701) for calculating bids and for purchase of goods/services with a plurality of auction entities; an algorithm (702) for selecting individual auction entities with which to trade; a web browser (506) to allow a user to search for auction entities and add these to a stored list of entities at the bidding device. Bid calculation and offer calculation algorithms operate to optimally place bids at a lowest overall cost to the trading entity, or place offers with the object of obtaining the highest overall monetary figures for goods or services for sale by the trading entity.

Fig. 7

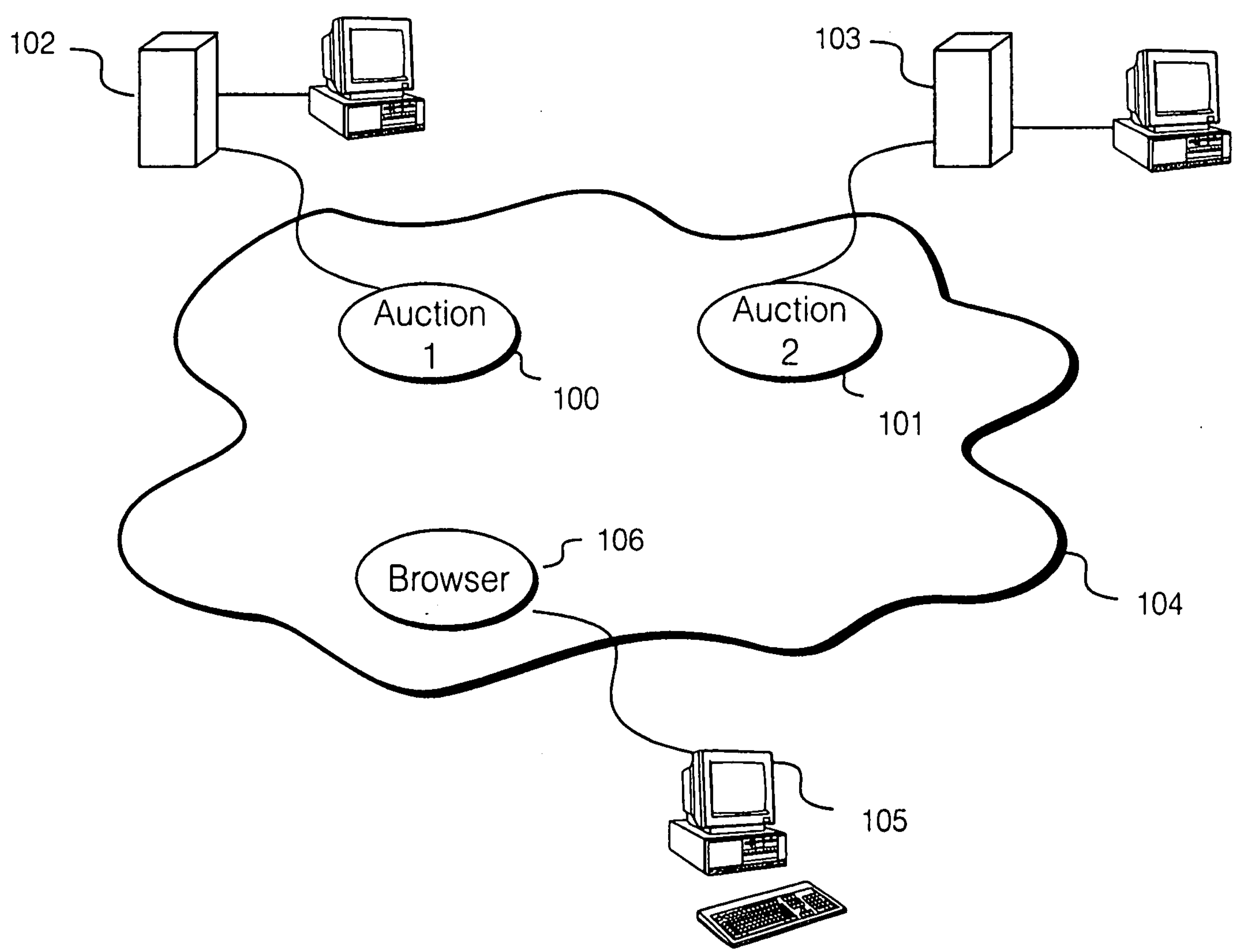


Fig. 1
(prior art)

Outstanding 2.88 Carat Round Diamond VVS2 I

Item #125854960

Jewelry, Gemstones:Gemstones:General

	Currently	$11,500.00 (reserve not yet met)	First bid	$5,000.00
	Quantity	1	# of bids	15 (bid history) (with emails)
Desc.	Time left	4 days, 20 hours +	Location	Portland, OR
	Started	07/01/99, 23:37:24 PDT		(mail this auction to a friend)
	Ends	07/11/99, 23:37:24 PDT		(request a gift alert)
			Featured Category Auction	
Bid!	Seller	Hendrik (206) (view comments in seller's Feedback profile) (view seller's other auctions) (ask seller a question)		
	High bid	Manolod (0)		
	Payment	Money Order/Cashiers Checks, personal Checks, See item description for payment methods accepted		
	Shipping	Seller ships internationally, See item description for shipping charges		

Seller assumes all responsibility for listing this item. You should contact the seller to resolve any questions before bidding. Currency is dollar ($) unless otherwise noted.

Fig. 2
(Prior Art)

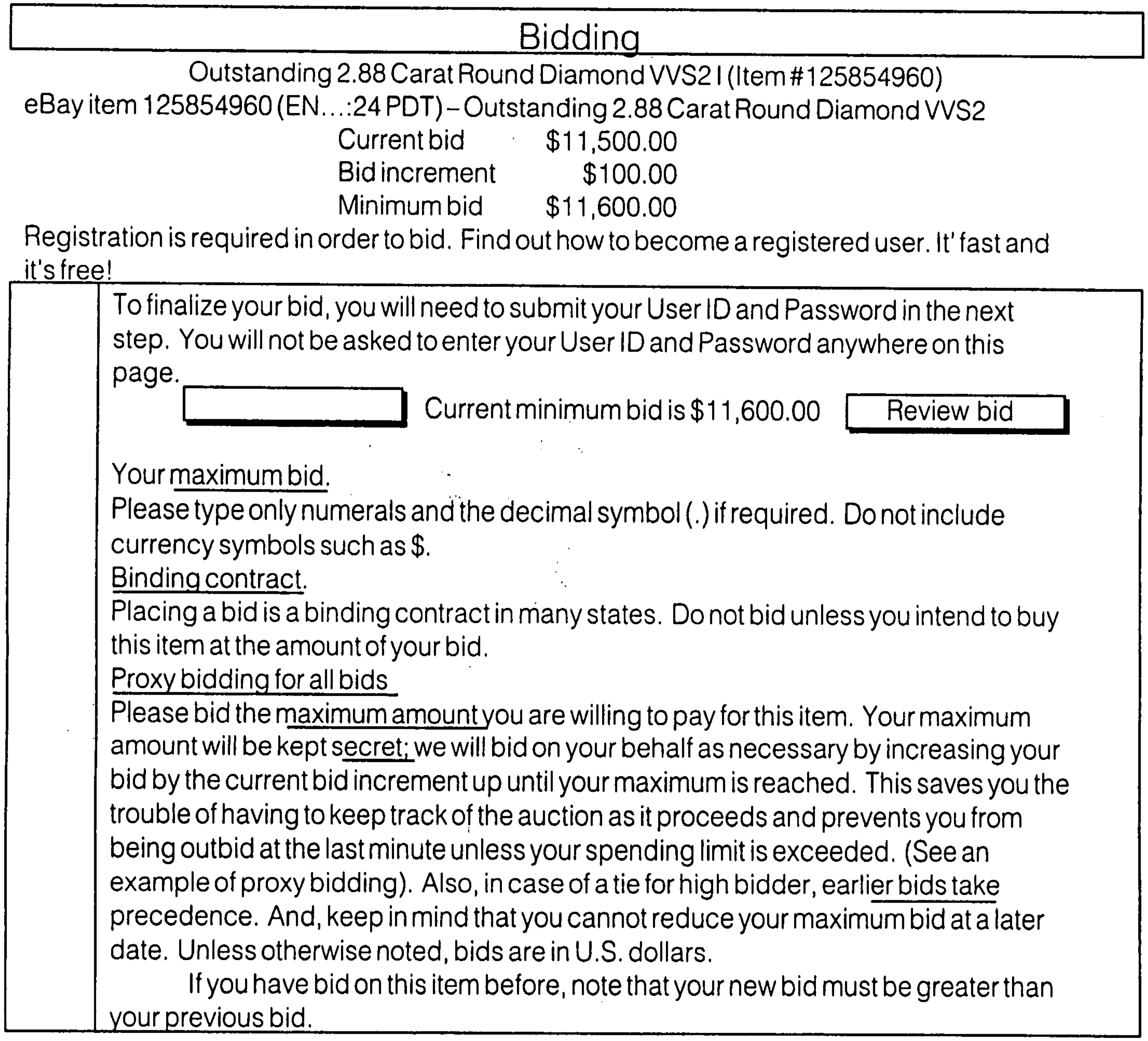


Fig. 3 (Prior Art)

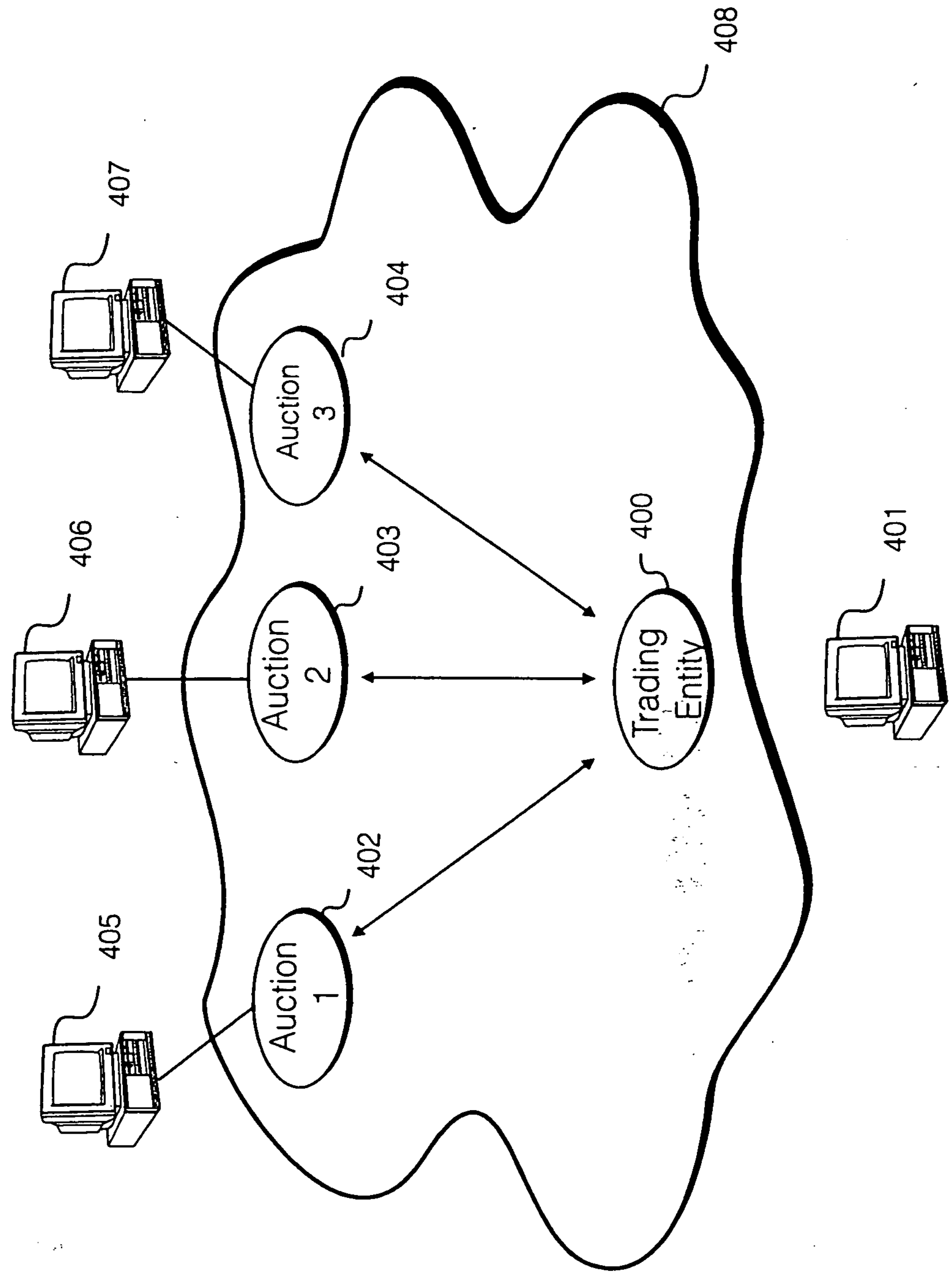

405
406
407
402
403
404
Auction 1
Auction 2
Auction 3
400
Trading Entity
401
408

Fig. 4

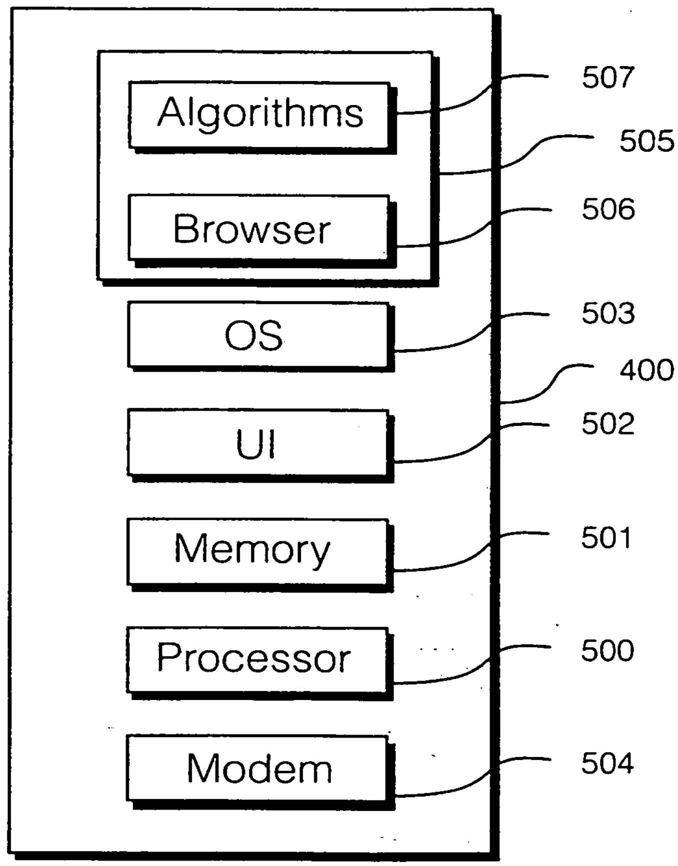

Algorithms
Browser
OS
UI
Memory
Processor
Modem
507
505
506
503
400
502
501
500
504

Fig. 5

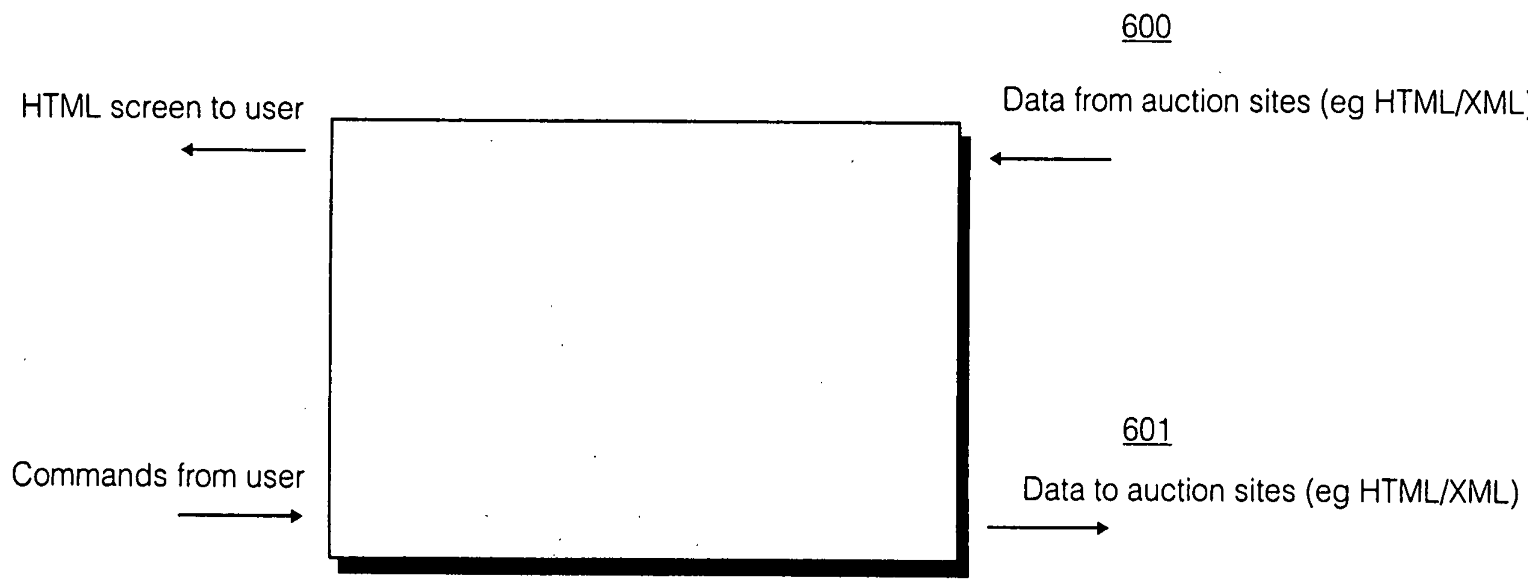

600
Data from auction sites (eg HTML/XML)
HTML screen to user
Commands from user
601
Data to auction sites (eg HTML/XML)

Fig. 6

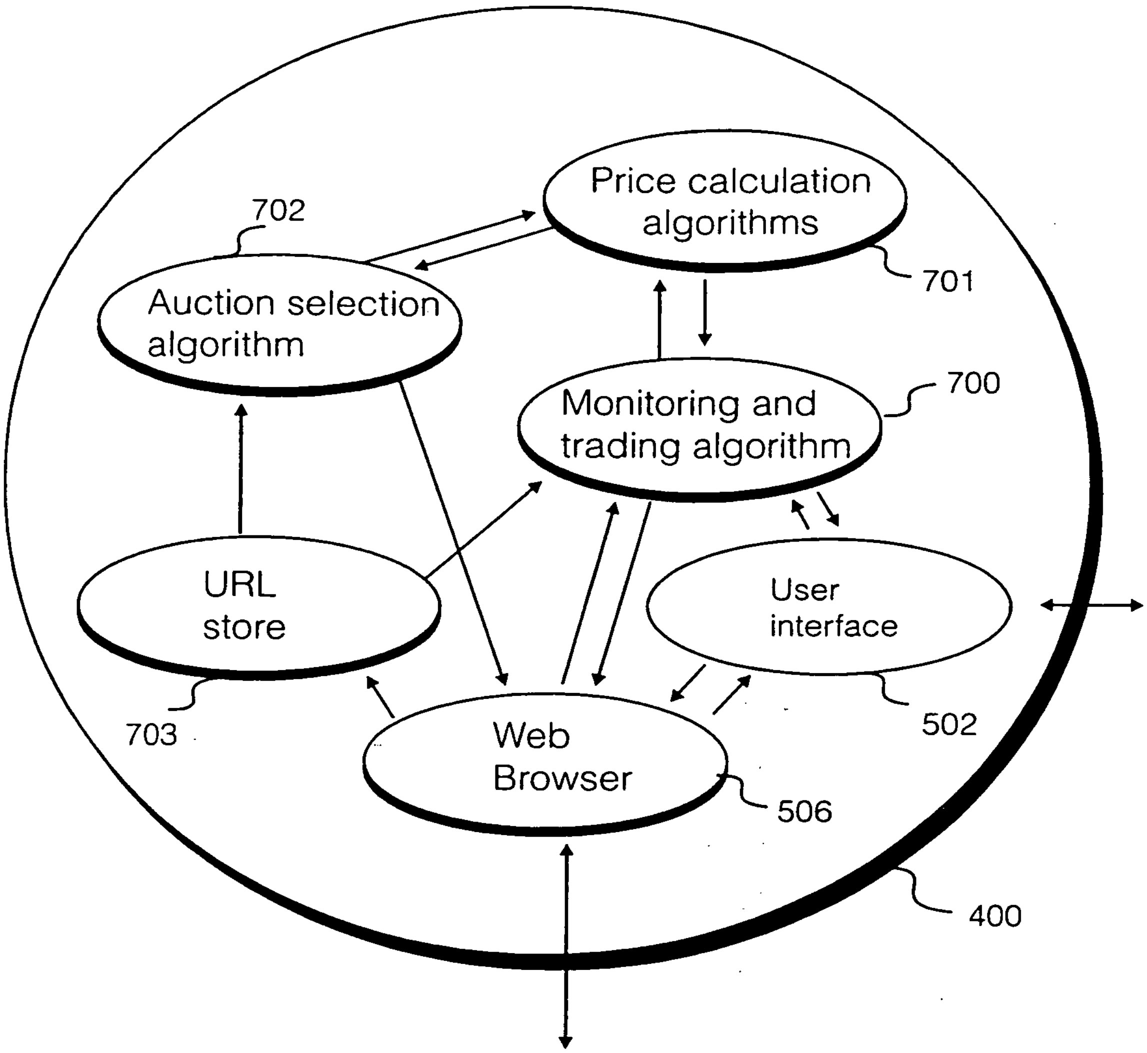

Price calculation algorithms
702
Auction selection algorithm
701
Monitoring and trading algorithm
700
URL store
User interface
703
Web Browser
502
506
400

Fig. 7

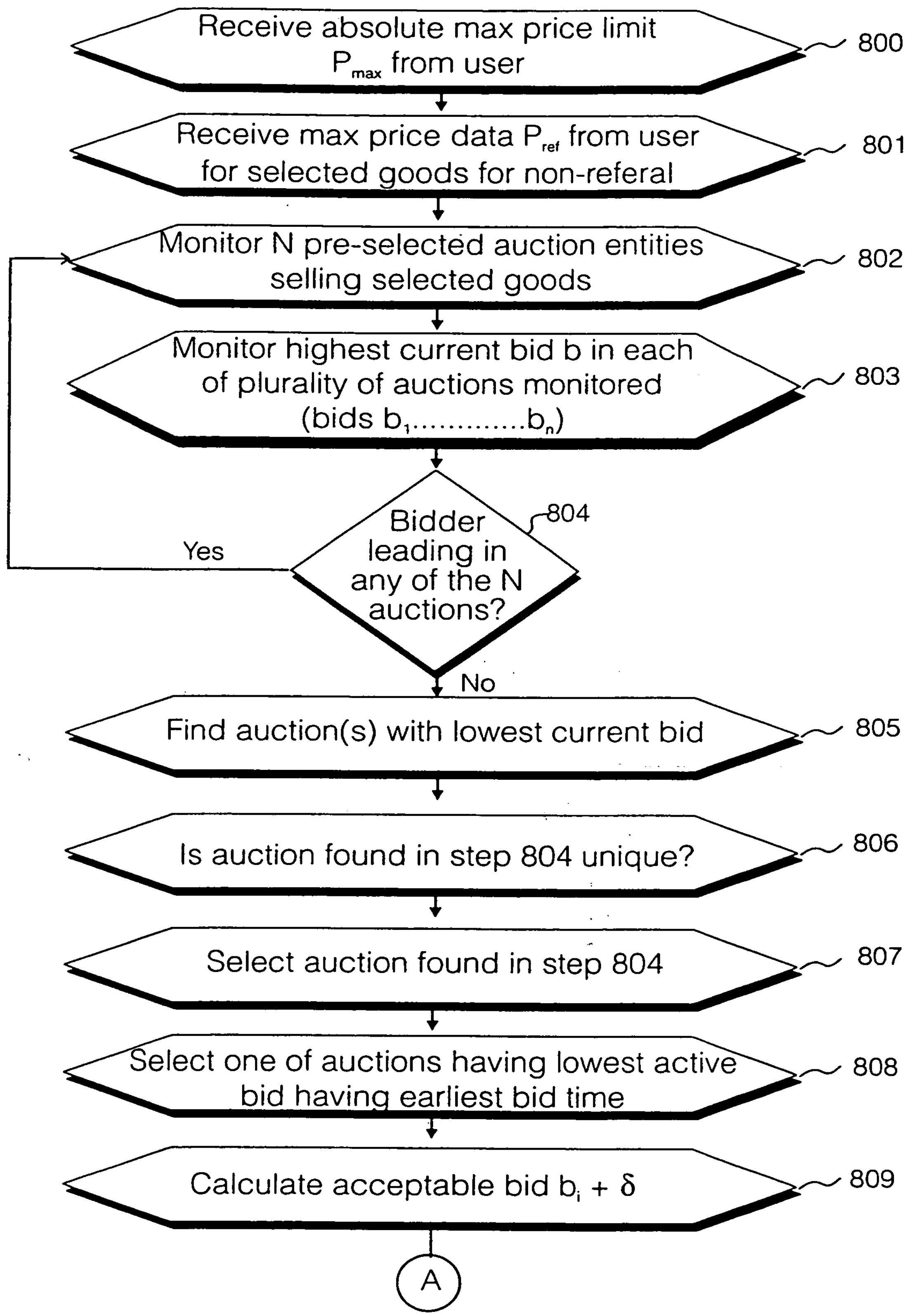

Receive absolute max price limit P_{max} from user
800
Receive max price data P_{ref} from user for selected goods for non-referal
801
Monitor N pre-selected auction entities selling selected goods
802
Monitor highest current bid b in each of plurality of auctions monitored (bids b_1............b_n)
803
Bidder leading in any of the N auctions?
804
Yes
No
Find auction(s) with lowest current bid
805
Is auction found in step 804 unique?
806
Select auction found in step 804
807
Select one of auctions having lowest active bid having earliest bid time
808
Calculate acceptable bid $b_i + \delta$
809
A

Fig. 8

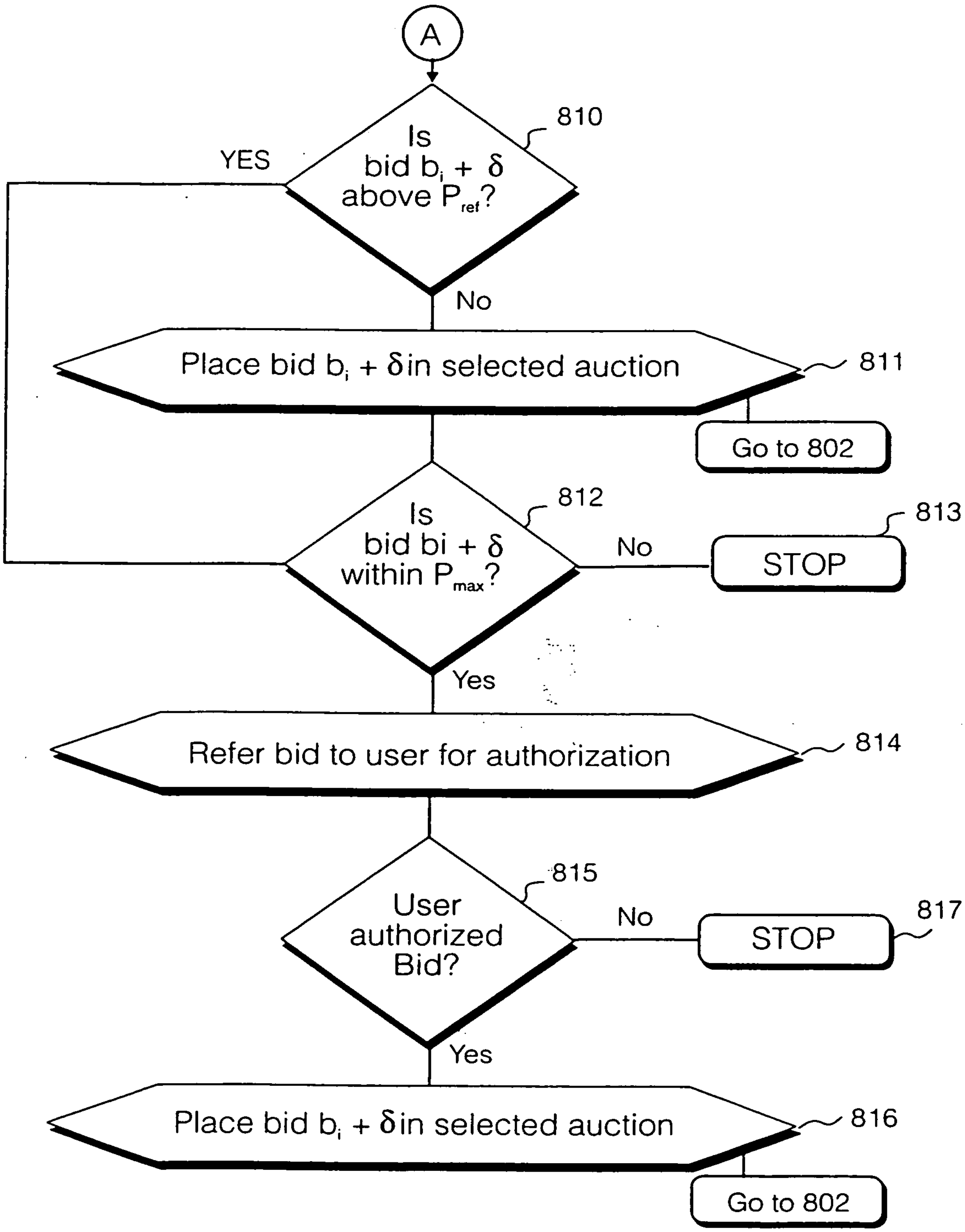

A
Is bid $b_i + \delta$ above P_{ref}?
810
YES
No
Place bid $b_i + \delta$ in selected auction
811
Go to 802
Is bid bi + δ within P_{max}?
812
No
STOP
813
Yes
Refer bid to user for authorization
814
User authorized Bid?
815
No
STOP
817
Yes
Place bid $b_i + \delta$ in selected auction
816
Go to 802

Fig. 8b

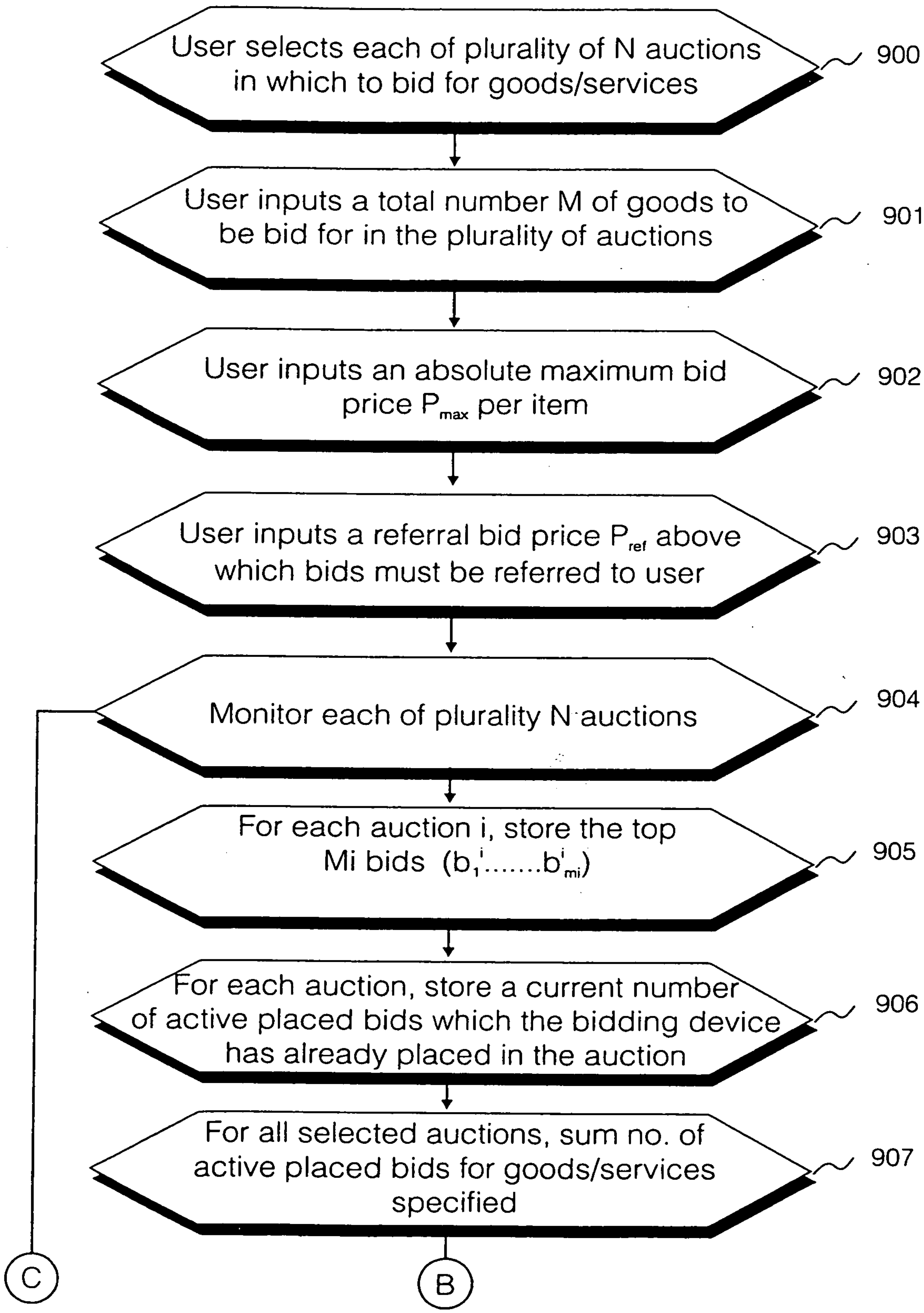

User selects each of plurality of N auctions in which to bid for goods/services
900
User inputs a total number M of goods to be bid for in the plurality of auctions
901
User inputs an absolute maximum bid price P_{max} per item
902
User inputs a referral bid price P_{ref} above which bids must be referred to user
903
Monitor each of plurality N auctions
904
For each auction i, store the top Mi bids $(b_1^i.......b_{mi}^i)$
905
For each auction, store a current number of active placed bids which the bidding device has already placed in the auction
906
For all selected auctions, sum no. of active placed bids for goods/services specified
907
C
B

Fig. 9

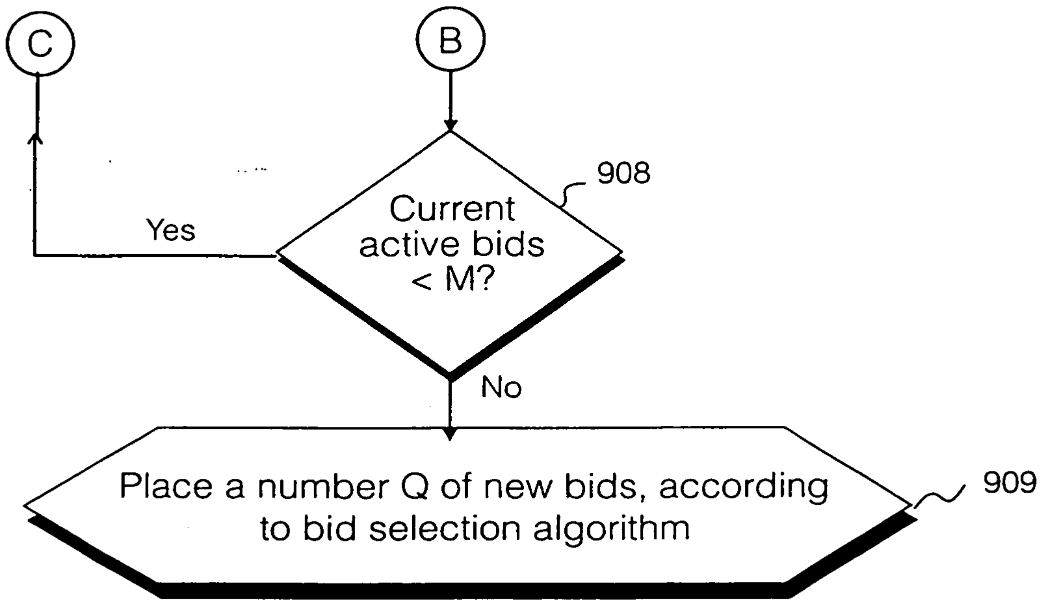

C
B
908
Current
active bids
< M?
Yes
No
Place a number Q of new bids, according
to bid selection algorithm
909

Fig. 9b

1000	1001	1002	1003	1004	1005	1006	1007	1008
Bid identifier	Auction ID	Lot No./ID	Goods/Services	Lot size (Units)	Amount bid (US Dollars)	Time of bid (GMT)	Close time (GMT)	Status
1278/99	372.278.596	1501	10k register	10,000	25 USD	10/2/99 -12.42	10/7/99 -17.00	Active
1279/99	281.502.7938	1720	10k register	10,000	27 Euro	10/2/99 -12.44	10/2/99 -18.00	Active
1280/99	282.702.6229	10	Dell monitor Model 1010	25	2500 USD	10/2/99 -12.50	10/2/99 -22.00	Active
1281/99	283.701.642	68	HP monitor 2050	25	3000 Euro	10/2/99 -12.59	11/2/99 -12.00	Active
1282/99	7298.73.25	1050231	Dell monitor	10	1100 Euro	10/2/99 -13.02	11/2/99 -12.00	Active
1283/99	25.9827.532	98710	Dell monitor	5	600 USD	10/2/99 -13.04		Active
1284/99	123.789.258	10/A99G	Lexus 200 GLS Automobile	1	25,001 USD	10/2/99 -15.02	15/2/99 -17.00	Outbid
1285/99	372.278.596	1502	10k register	10,000	27 USD	10/2/99 -15.03	10/2/99 -17.00	Active
1286/99	372.278.596	1503	10k register	10,000	28 USD	10/2/99 -15.03	10/2/99 -17.00	Active
1287/99	281.502.7938	1724	10k register	10,000	28 USD	10/2/99 -15.04	10/2/99 -18.00	Active

Fig. 10

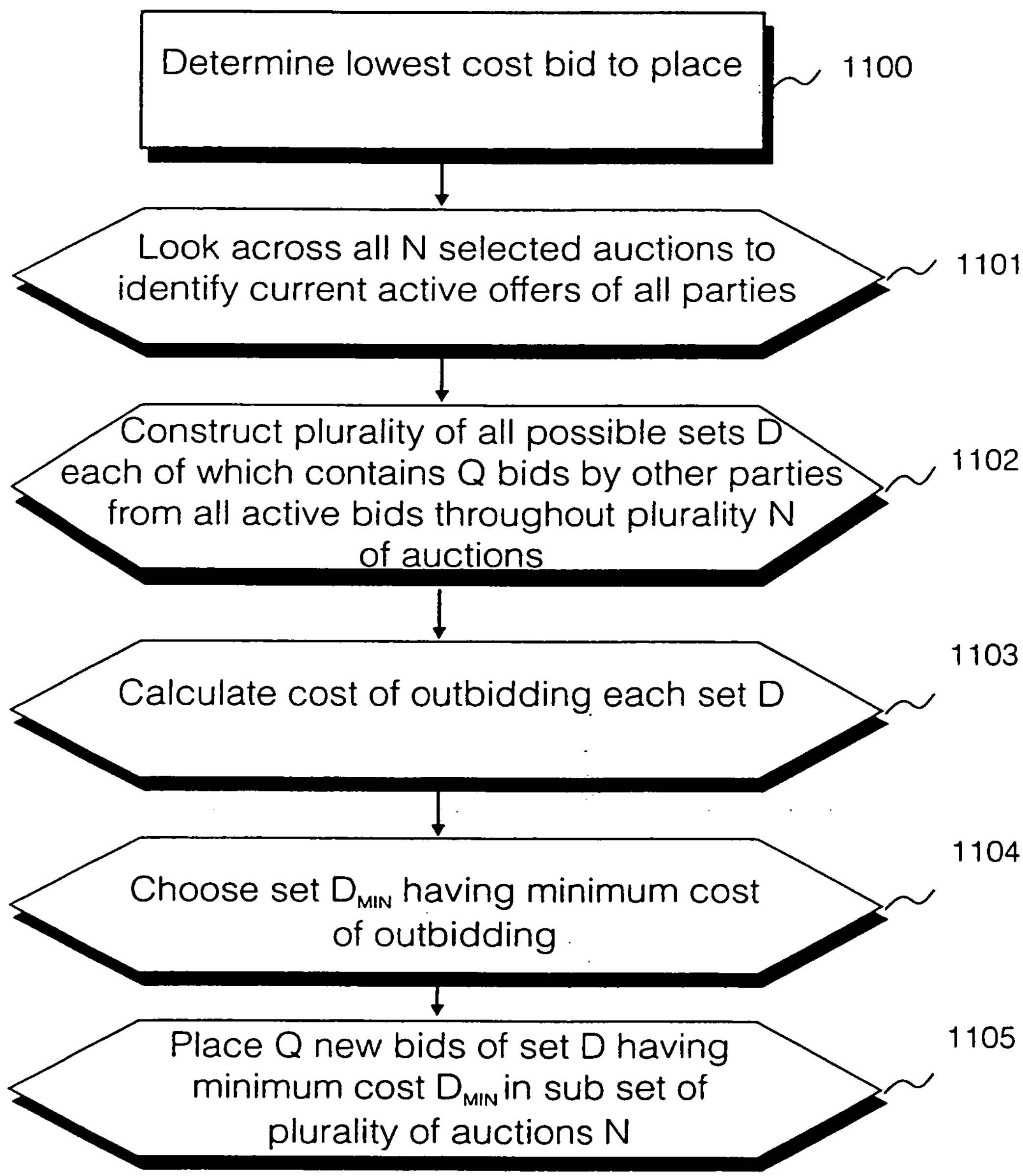
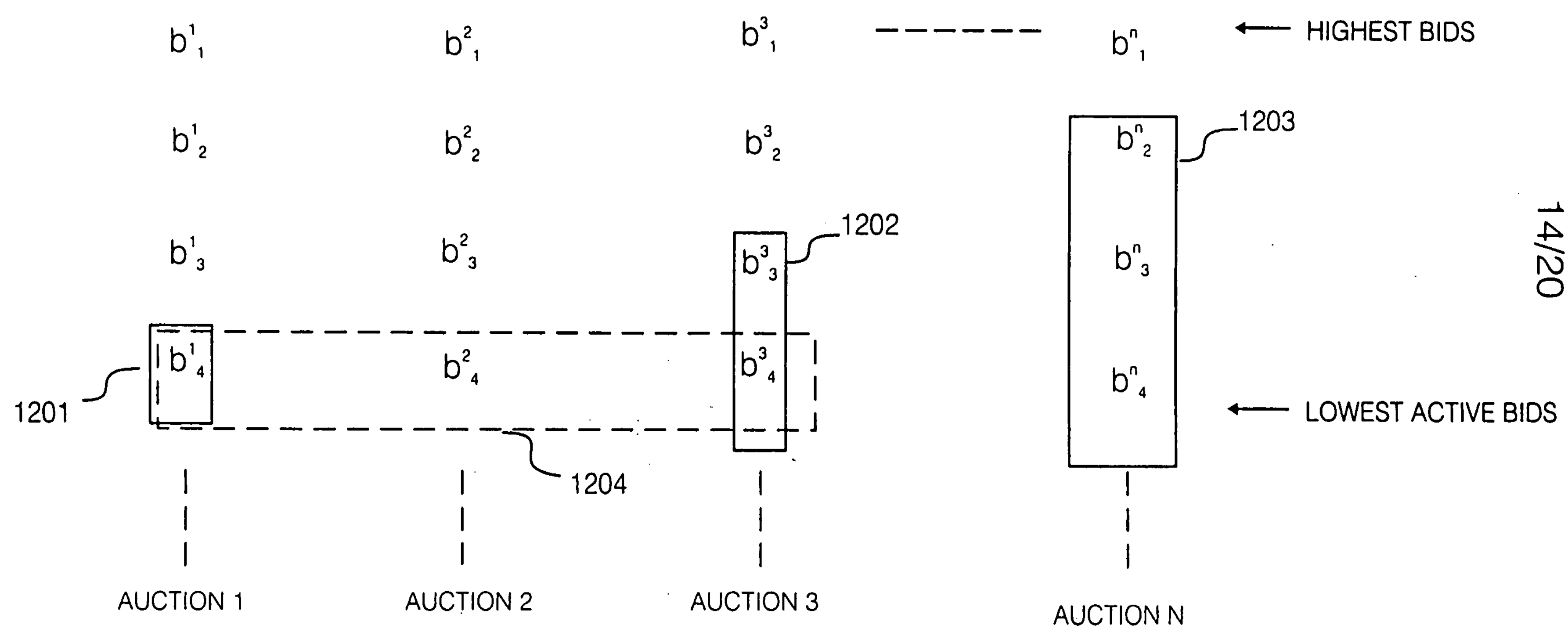

Determine lowest cost bid to place
1100
Look across all N selected auctions to identify current active offers of all parties
1101
Construct plurality of all possible sets D each of which contains Q bids by other parties from all active bids throughout plurality N of auctions
1102
Calculate cost of outbidding each set D
1103
Choose set D_{MIN} having minimum cost of outbidding
1104
Place Q new bids of set D having minimum cost D_{MIN} in sub set of plurality of auctions N
1105

Fig. 11



HIGHEST BIDS
LOWEST ACTIVE BIDS
b^1_1
b^1_2
b^1_3
b^1_4
b^2_1
b^2_2
b^2_3
b^2_4
b^3_1
b^3_2
b^3_3
b^3_4
b^n_1
b^n_2
b^n_3
b^n_4
1201
1202
1203
1204
AUCTION 1
AUCTION 2
AUCTION 3
AUCTION N

Fig. 12

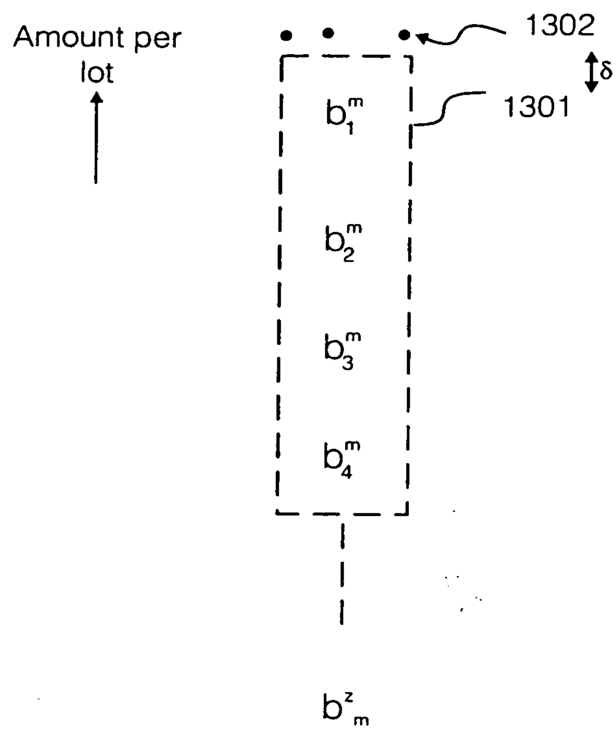


Fig. 13

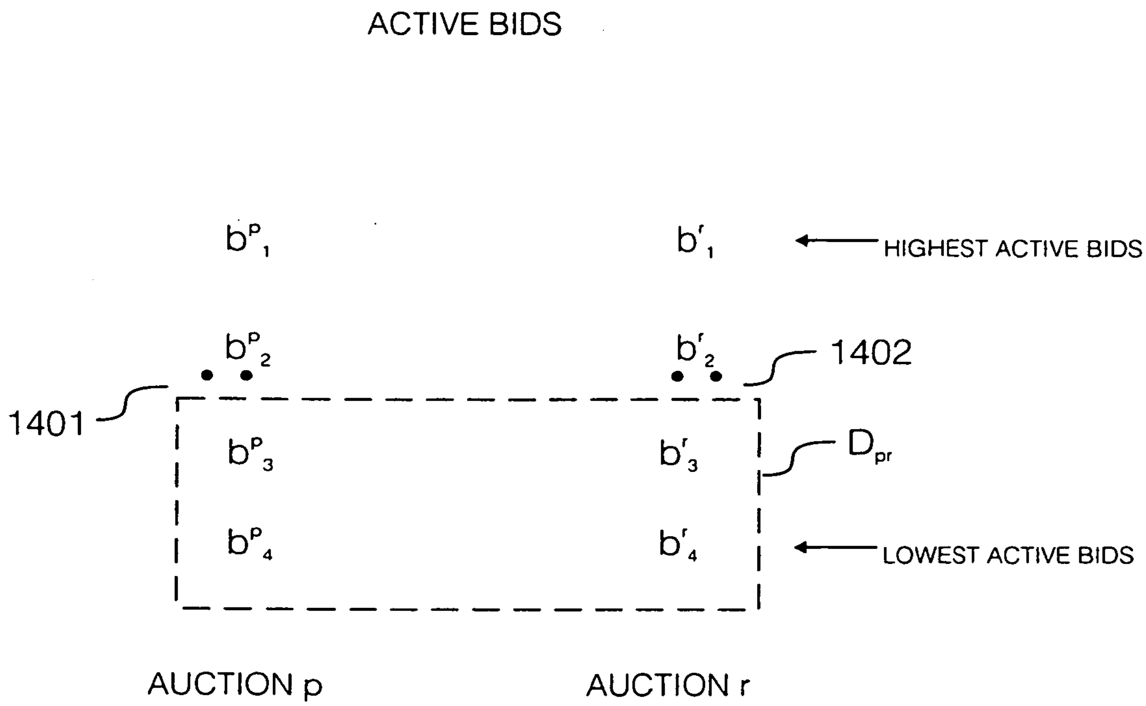

ACTIVE BIDS
b^p_1
b^r_1
HIGHEST ACTIVE BIDS
b^p_2
b^r_2
1402
1401
b^p_3
b^r_3
D_{pr}
b^p_4
b^r_4
LOWEST ACTIVE BIDS
AUCTION p
AUCTION r

Fig. 14

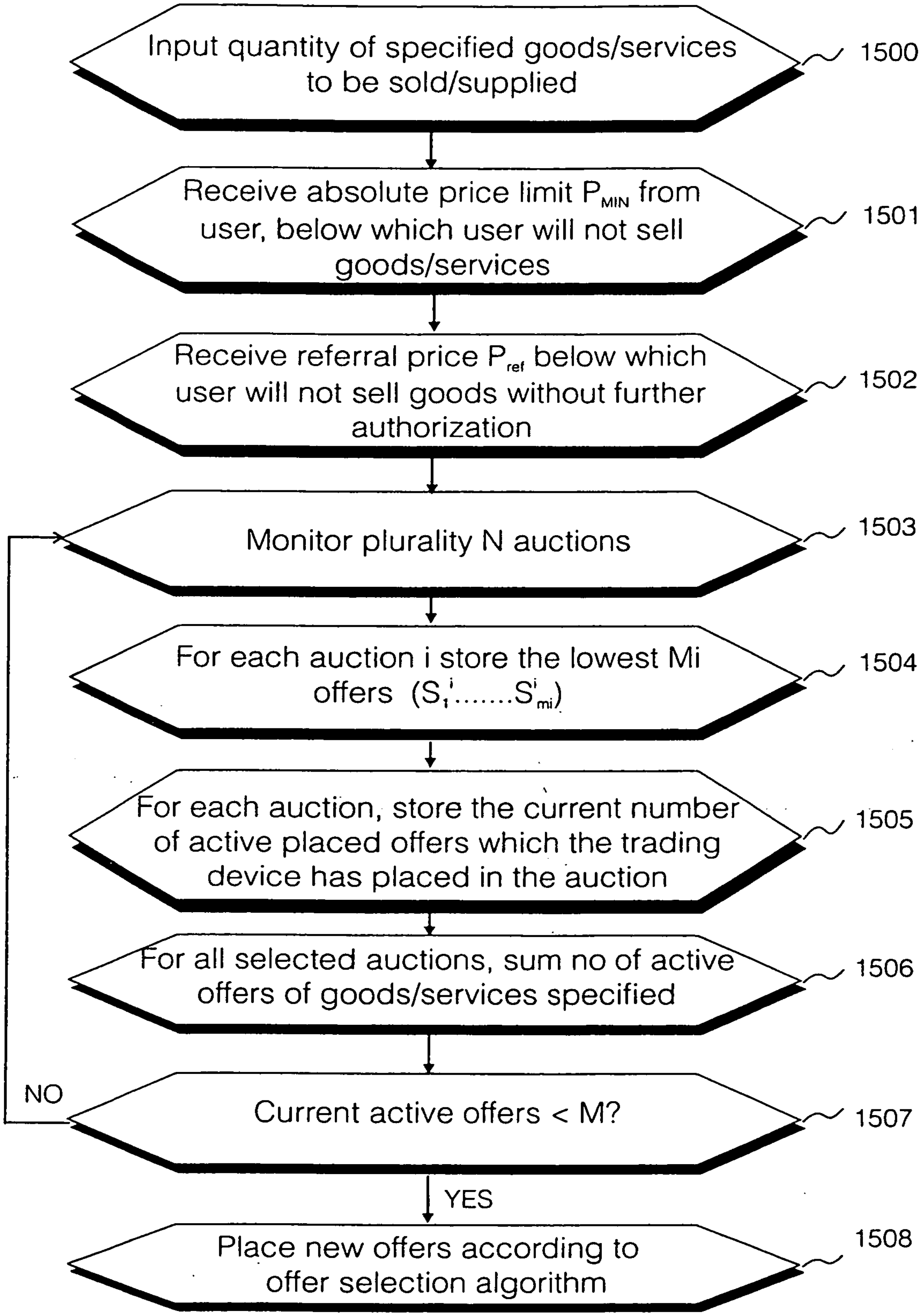

Input quantity of specified goods/services to be sold/supplied
1500
Receive absolute price limit P_{MIN} from user, below which user will not sell goods/services
1501
Receive referral price P_{ref} below which user will not sell goods without further authorization
1502
Monitor plurality N auctions
1503
For each auction i store the lowest Mi offers $(S_1^i.......S_{mi}^i)$
1504
For each auction, store the current number of active placed offers which the trading device has placed in the auction
1505
For all selected auctions, sum no of active offers of goods/services specified
1506
NO
Current active offers < M?
1507
YES
Place new offers according to offer selection algorithm
1508

Fig. 15

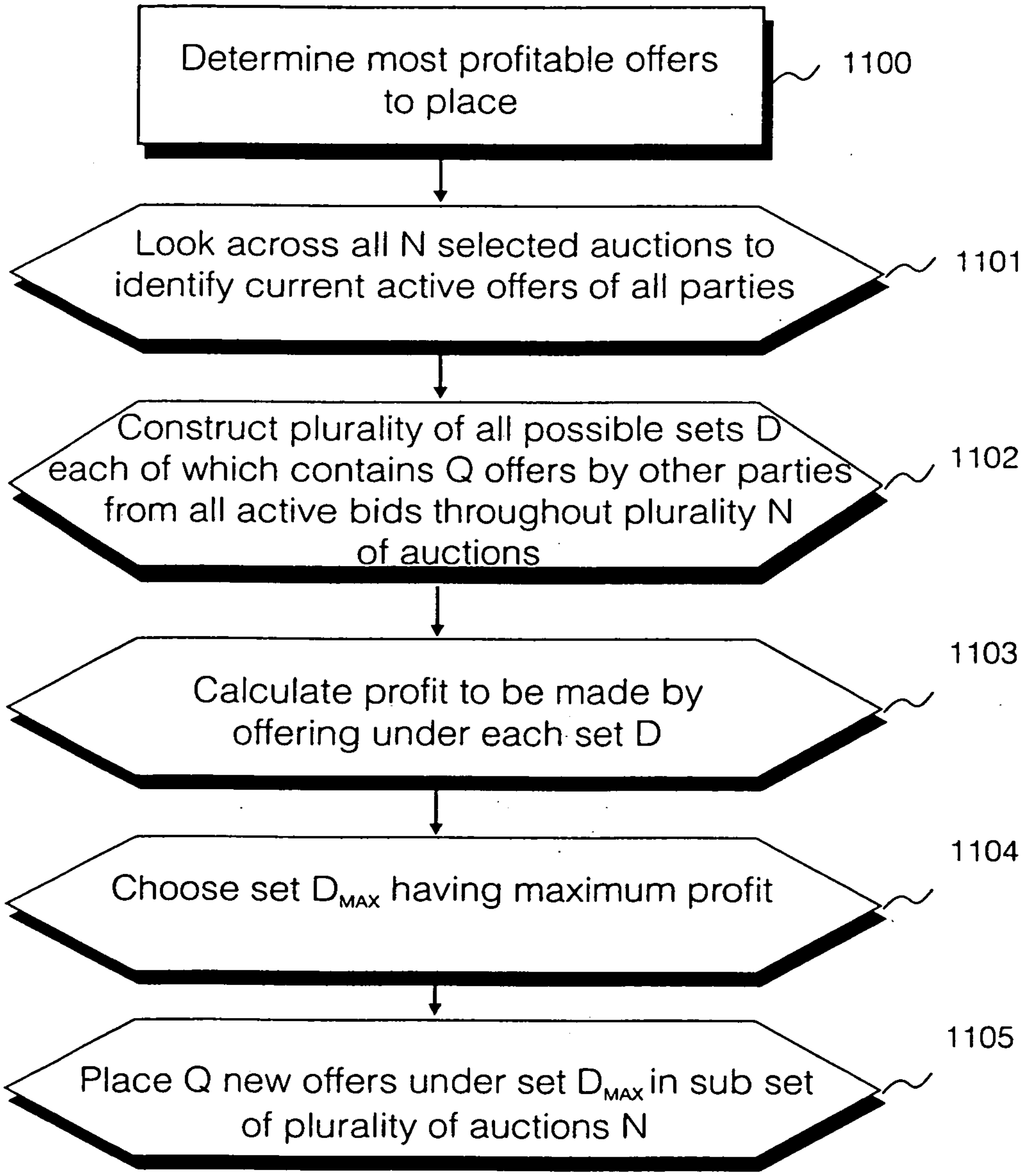

Determine most profitable offers to place
1100
Look across all N selected auctions to identify current active offers of all parties
1101
Construct plurality of all possible sets D each of which contains Q offers by other parties from all active bids throughout plurality N of auctions
1102
Calculate profit to be made by offering under each set D
1103
Choose set D_{MAX} having maximum profit
1104
Place Q new offers under set D_{MAX} in sub set of plurality of auctions N
1105

Fig. 16

Auction 1 bids	Auction 2 bids	Auction 3 bids		Auction n bids
b^1_1	b^1_2	b^1_3	------------	b^1_n
b^2_1	b^2_2	b^2_3		b^2_n
b^3_1	b^3_2	b^3_3		b^3_n
⋮	⋮	⋮		⋮
b^z_1	b^z_2	b^z_3	------------	B^z_n

Fig. 17

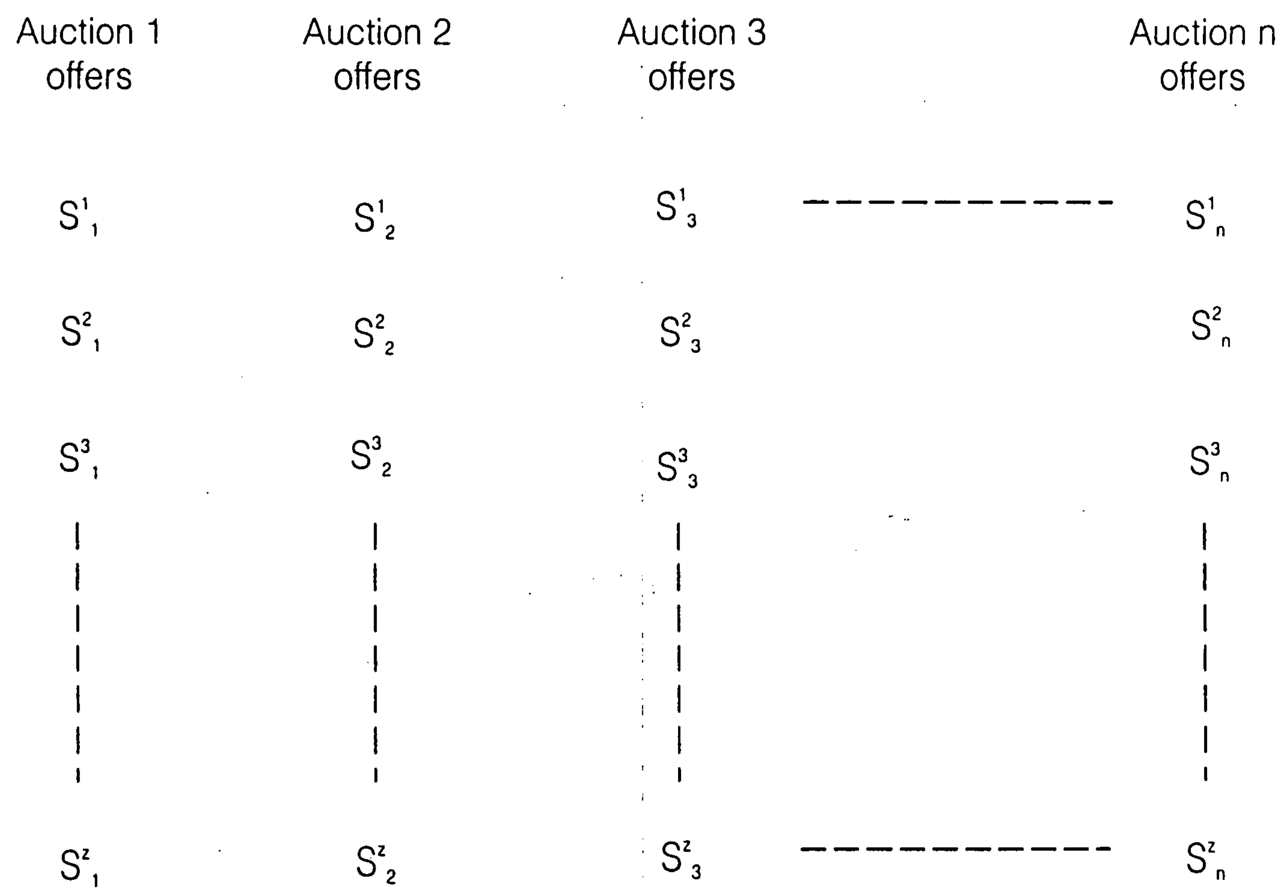

Auction 1 offers
Auction 2 offers
Auction 3 offers
Auction n offers
S^1_1
S^1_2
S^1_3
S^1_n
S^2_1
S^2_2
S^2_3
S^2_n
S^3_1
S^3_2
S^3_3
S^3_n
S^z_1
S^z_2
S^z_3
S^z_n

Fig. 18

Attorney Docket No. 30990046

THIS PAGE BLANK (USPTO)